

The International Journal of Digital Curation

Issue 3, Volume 4 | 2009

Emulation: From Digital Artefact to Remotely Rendered Environments

Dirk von Suchodoletz,

University of Freiburg

Jeffrey van der Hoeven,

Koninklijke Bibliotheek

Summary

Emulation used as a long-term preservation strategy offers the potential to keep digital objects in their original condition and experience them within their original computer environment. However, having just an emulator in place is not enough. To apply emulation as a fully fledged strategy, an automated and user-friendly approach is required. This cannot be done without knowledge and contextual information of the original software. This paper combines the existing concept of a view path, which captures the contextual information of software, together with new insights into improving the concept with extra metadata. It provides regularly updated instructions for archival management to preserve and access its artefacts. The view-path model requires extensions to the metadata set of the primary object of interest and depends on additionally stored secondary objects for environment recreation like applications or operating systems. This article also addresses a strategy of rendering digital objects by running emulation processes remotely. The advantage of this strategy is that it improves user convenience while maximizing emulation capability.¹

¹ This article is based on the paper given by authors at iPRES 2008; received November 2009, published December 2009.

The *International Journal of Digital Curation* is an international journal committed to scholarly excellence and dedicated to the advancement of digital curation across a wide range of sectors. ISSN: 1746-8256 The IJDC is published by UKOLN at the University of Bath and is a publication of the Digital Curation Centre.



Challenges in Long-term Preservation

Unlike books, newspapers, photographs or other traditional material, digital objects require a digital context consisting of a combination of software and hardware components. Due to technological advances, hardware and software become obsolete, leaving us uncertain as to whether we will still be able to render today's digital objects in the future. Permanent access to archived digital artefacts thus raises challenges to archive operators who have to deliver continuing access to digital material without loss of information.

Several solutions for long-term access exist, of which migration and emulation may be regarded as the chief protagonists. Migration – the most widely used digital archiving strategy today – seeks to address this problem by changing the digital object in order to prepare it for access and rendering in future digital environments. Although this strategy applies to static digital objects such as images, text, sound and animation, it is not suitable for dynamic objects such as educational software or computer games. As a lot of digital material is becoming more advanced, relying solely on migration as a preservation strategy is risky and will certainly result in loss of authenticity and information.

Emulation offers a different approach. It does not change the digital object itself, but tries to recreate the original computer environment in which the object used to be rendered. Each layer of the software-hardware-stack can be used as a working point for emulation: applications, operating systems or hardware can be recreated in software by using an emulator for the actual environment.

However, an emulator also relies on a computer environment. From the perspective of archive management, emulators do not differ significantly from other digital objects. Even emulators become obsolete with the evolution of digital environments. Several strategies exist to keep the emulators available in a changing environment (Verdegem & Van der Hoeven, 2006). For example, the Koninklijke Bibliotheek (KB) and Nationaal Archief of the Netherlands developed Dioscuri,² an x86 emulator developed for the purpose of long-term archiving, took account of Van der Hoeven, Lohman and Verdegem (2008). This emulator bridges the widening gap between older x86 machinery and recent architectures by using a virtual layer between operating system and emulator to abstract from specific reference platforms. Furthermore, detailed documentation on every step taken in design and development is being preserved to allow future users and developers to understand the software.

Bridging the Past to the Future

No matter which emulator is chosen, there is always a need for contextual information on the computer environment involved. For example, questions such as “for which operating systems is WordPerfect 5.1 compatible?” are less obvious today than they were 20 years ago. To overcome this gap of missing knowledge, a formalization process is needed to compute the actual needs for an authentic rendering environment of the digital artefact. In 2002, IBM Netherlands proposed the concept of a *view-path* based on their *Preservation Layer Model (PLM)* (van Diessen, 2002) which has been refined during the research on emulation at Freiburg University and the European project Planets.

² Dioscuri, the durable x86 emulator in Java <http://dioscuri.sourceforge.net>

The PLM outlines how a file format or collection of similar objects depends on its environment. A PLM consists of one or more layers of which each layer represents a specific dependency. The most common PLM consists of three layers: application layer, operating system layer and hardware layer. However, other variations can also be created. Based on a PLM, different software and hardware combinations can be created. Each such combination is called a view path. In other words, a view path is a virtual line of action starting from the file format of a digital object and linking this information to a description of required software and hardware. Figure 1 illustrates some typical view paths starting from a particular digital object. Depending on the type of object a specific rendering application is required. This application requires a certain operating system (OS) to be executed, while the OS, in turn, relies on particular hardware.

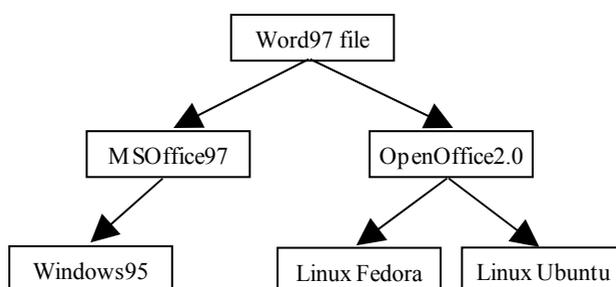


Figure 1. Example view paths.

As dependencies might change in the future, the view paths originally derived may also change over time. This is indeed the case when certain hardware and software become obsolete. To solve this missing link, the dependency can be replaced by another compatible environment or by using emulators to bridge the gap between the digital past and future (Figure 2).

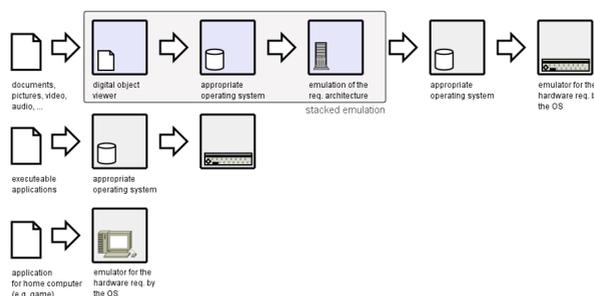


Figure 2. Emulation and view paths.

Looking into the future, the following situations may arise in respect of object dependencies:

- At a given point in time there is exactly one view path for an object to its rendition.
- An object has become inaccessible because all view paths have become obsolete.
- There are several different view paths for a digital object available which require a selection procedure.

The first situation leaves no room for discussion as there is only one way to retain access to the digital object. The second situation requires some additional processing since one or more layers of the view path have become obsolete. This can

be solved by using emulators instead.

The third situation however is not easily resolved. To manage various rendering options, a procedure will be required to find the most preferred view path for rendering a certain object or collection of objects.

View Path Extensions Using Metrics

To apply the PLM in combination with emulation in archival management a formalization and automation of the decision process is required. To do so, the model can be extended with metrics. A metric could be any kind of measurement along a view path and can be created by attaching a certain weight on a subsection of a view path. Current metadata for a layer in a view path need to be extended to capture metric information. Having applied weightings to all view paths, a classification can be devised as to what the metric stands for. In general, using view path metrics offers the following advantages (see also Figure 3) since they:

- allow comparison of each option to ensure a high grade of authenticity and quality of the object rendering or execution;
- offer quantifiers to emphasize particular aspects, such as authenticity or ease of use;
- include the archive users' preferences with respect to applications, operating systems or reference platforms;
- allow cost-benefit analysis, quantifying which view paths are economically feasible and which are not.

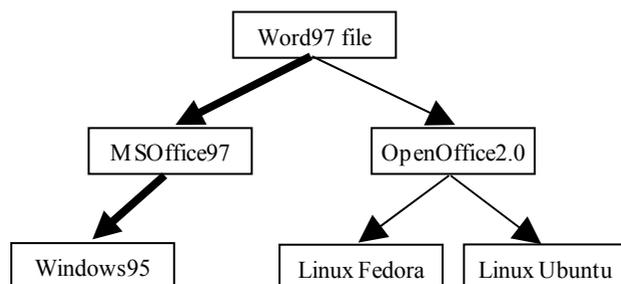


Figure 3. Example view paths with weighting denoted by arrow.

Assigning weights to a view path based on the authenticity of a certain computer environment can help a user find the most authentic representation of a digital object. Moreover, weights could be altered to influence a requested rendering in a certain direction. Furthermore, users of computer environments can help to evaluate view paths by adding reviews and ratings to a view path based on quality, completeness and correctness, or ease of use.

Another way in which view path metrics can be helpful is managing the costs of preserving the original environment. During the preservation period of each digital object the determined view path for the associated object type has to be checked on every change of the reference environment. For example, obsolete hardware or updates for software affect the object's dependencies and should therefore be considered in the associated view path as well. Furthermore, changing the view path also requires changes in the actual emulation environment which results in various updates in hardware, software and configurations.

Maintenance of each view path and environment inevitably attracts certain costs. Attaching metrics representing operational costs to each view path can deliver an estimate of the effort required to archive a specific type of object. If these costs pass a certain threshold, economic considerations could be taken into account when ingesting the objects into the archive. This knowledge may help to inform decisions to be made on preferred formats or specific types of objects over others.

If multiple view paths exist for a given object type, their costs would offer another metric to decide on sound alternatives. For example, assume a digital object is formatted according to the PDF 1.0 Standard and that it is accessible with a tool for MS Windows 3.11. If there are other tools offering the same results in quality and authenticity, and there are no other object types requesting this specific view path, it might be advisable to drop this environment and aggregate the paths. This not only lowers administration costs for keeping the view path current, but also reduces the costs of preserving the necessary software in an archive.

Digital Archive Management

From the emulation perspective, several tasks have to be carried out to ensure that a digital object is preserved and accessible in a long-term digital archive. In general, three phases can be distinguished: the required workflow steps on object ingest; the periodical operational procedures of archive operation; and the procedures for the object digest to the interested user of a digital object.

Identification and characterisation of digital objects have to be carried out on ingest. Several solutions already exist, of which the most prominent at the moment are PRONOM and DROID of The National Archives in the UK³. Although these tools are able to offer information about the digital format and some of its dependencies, they do not however take into account all computer-related dependencies such as hardware and emulators. Therefore, extensions should be made to incorporate the PLM and its extensions for keeping track of metrics in the model.

Another important part of archive management is the selection of proper emulators. At ingest time and during the whole period of preservation, availability of view paths has to be checked and if a view path has become obsolete, emulators can be used to close the gap between the layers in the path. If no suitable view path can be constructed, the digital object may have to be rejected at ingest time because no guarantee can be given that it will remain accessible over the long term.

Furthermore, even when a view path holds emulator and contextual information, there may still remain certain implications at the time a digital object is disseminated and needs to be rendered. Firstly, the original environment consisting of software and hardware must be preserved. Secondly, an emulation service is required to reconstruct the original environment, configure the emulator, and activate the emulation process. These two aspects will be discussed in more detail in the following sections.

³ The National Archives (TNA): PRONOM <http://www.nationalarchives.gov.uk/pronom/> & DROID <http://droid.sourceforge.net/>

Software Archive

In recent years, a lot of attention has been paid to emulation and virtualisation software as the primary requirement for retaining access to any kind of digital information while assuring authenticity. However, emulators only solve one part of the equation. Additional software such as the operating system and applications are also required (Reichherzer & Brown, [2006](#)).

No standardized or coordinated approach for software preservation currently exists. Some national libraries treat software releases the same way as they do publications, and preserve them on the shelf next to their books and journals⁴. Although these software releases are indexed and managed, the actual bits are still on their original media carriers and are not directly accessible by library visitors. Media deterioration is a serious threat and might result in loss of information in the near future. Other organisations, such as the KB or University of Freiburg, rely on external sources such as software companies to handle the preservation of released software.

In order to understand better why this area is as yet somewhat neglected, one must take account of several hindrances to the preservation and access to software. Firstly, the newer a computer environment is, the higher the level of complexity pertains and the greater the number of additional software components needed. Current computer systems are running very complex applications which rely on a wide range of utilities such as hardware drivers, plug-ins, video and audio decoders, fonts and many more. Preserving such complex applications implicitly requires all dependent sources to be preserved as well.

Secondly, legal issues arise. Digital rights management and copy protection mechanisms can prevent one from copying the original bit stream from its carrier into a digital archive. Even if it is technically possible to preserve it, legal implications may still pertain that may prevent future generations from using the software. To complicate matters, some software requires an online activation or regular updates in order to remain operational. Having the software package alone may not be sufficient for future usage.

A third issue is to understand how software operates. This might be obvious today, but can become problematic in the future. Extensive metadata are required to deal with this problem, addressing not only the title and release date of software but also more semantic information such as installation manuals, tutorials and reference guides.

A final obstacle is the diversity of software releases. Most software is adapted to different human languages, geographical areas and units of different parameters. The latter include various currencies, their representation being effected in varying specific characters, dimensions or sizes, not to mention format of date ('date syntax'), calculations and even the number of public or religious holidays.

These difficulties aside, the original software also needs to be preserved if digital objects are to be kept alive via emulation. Guidelines similar to those created for digital objects themselves must be brought to bear in order to safeguard emulators,

⁴ Bibliothèque nationale de France (BnF), <http://www.bnf.fr>

operating systems, applications and utilities. That is, software should be stored under the same conditions as other digital objects by preserving them in a OAIS-based (ISO 14721:2003) digital archive.

Nevertheless, it might be of interest to retain various access copies of software to allow emulators to prepare them for convenient use during the realization of a view path. Requested view paths could often be stored as combined caches of applications, operating systems and the emulator in order to provide faster access. It would be useful to distribute such specifically prepared containers between memory institutions in order to share out the management costs and overheads.

Remote Access to Emulated Environments

Assuming that the required software and metadata are available for emulation, it is necessary to prepare the environment to be emulated. This is a very technical process and requires skilled personnel to merge all required software into one computer environment, set emulator parameters and offer guidance to the user about how to work with ancient computer environments.

To tackle this challenge it would be desirable to centralize the whole process in specialized units with trained personnel and offer services within a framework over the Internet. This reduces the complexity of the procedures required to run emulators as well as the system requirements of the users' viewer, preferably a web browser. The users receive the results presented remotely via a virtual screen on their computer. In overview, this kind of setup would offer the following benefits:

- Access to digital objects is location-independent.
- No special system requirements on the user's side are necessary.
- Management of such a service can be centralized and several memory institutions could share the workload or specialize in certain environments and share their expertise with others.
- Problems of licence handling and digital rights management could be avoided, because software does not need to be copied onto users' private machines, instead merely being run by the service provider.
- Organisations such as computer museums are able to present their collections in an alternative way as they are no longer restricted to one room.

Nonetheless, knowledge about old computer environments is necessary in order to be able to work with emulated computers, but on-screen instructions might offer users extra assistance.

The Planets Project is carrying out a pilot which is developing a prototype of an emulation service. This service is based on existing emulators such as Dioscuri and allows them to run on a remote basis. Transportation of the remotely rendered environment is done by GRATE (Global Remote Access to Emulation Services) and is currently being developed by the University of Freiburg. With GRATE any user can easily access emulated environments remotely via their web browser.

Initial experiments have indicated that this solution is very user-friendly and flexible in its configuration. Figures 4 and 5 show two screenshots of GRATE. The

first one runs Dioscuri remotely to load WordPerfect 5.1. The second image shows the desktop of Windows98 being executed by the QEMU emulator⁵. Both emulated environments are accessed via a normal web browser.

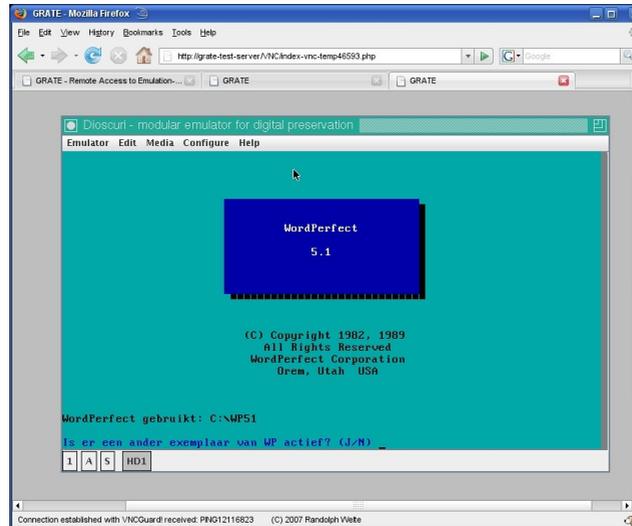


Figure 4. GRATE running Dioscuri with WordPerfect5.1.

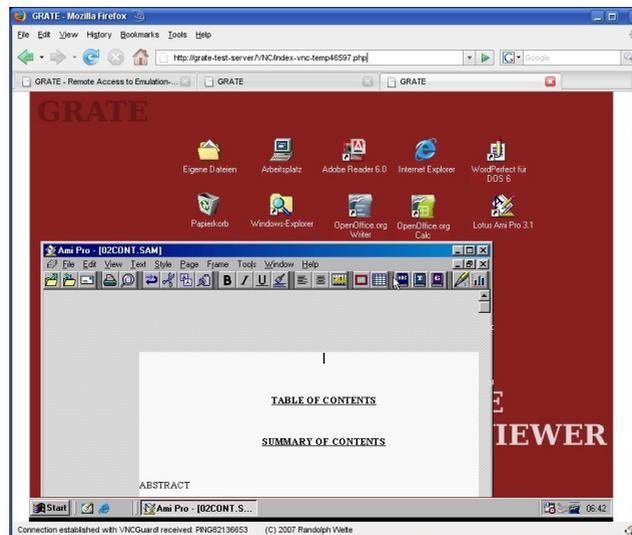


Figure 5. GRATE running QEMU with Windows9.

The next step is to integrate this emulation service with the interoperability framework of Planets. This will result in a major extension to functionality for preservation action strategies thereby allowing a Planets-user to start emulation activities automatically when a digital object needs to be rendered in its authentic computer environment.

Conclusions

Emulation strategies help to offer sustainable access to digital objects in their authentic environment. Aside from an emulator, other conditions have to be met to assure the successful rendering of the object. Specific information about the object's

⁵ QEMU website <http://bellard.org/qemu/>

dependencies on hardware and software should also be preserved. Furthermore, access to the original software is necessary to recreate an old computer environment, and an access mechanism is also required for configuring the emulator and environment.

Use of the Preservation Layer Model (PLM) represents a flexible solution to the management of metadata relating to environmental dependencies. The PLM introduces view paths for each combination of hardware, software and digital file format or collection of files. These formalizations can assist archivists in managing their digital collections and provide them with guidelines on what to do on object ingest, during object storage, and on dissemination. However, the current PLM structure does not explain how to proceed when multiple view paths can be applied. To overcome this difficulty, view path metrics may help to optimize this operation by attaching weights to each layer of dependencies. These weights can be influenced by several factors such as “most commonly used operating system” or “user-preferred application”. The community could even have a vote on the selection procedure by offering feedback and ratings. Moreover, a cost/benefit analysis can be applied to drop less effective view paths.

The original software is needed in addition to the metadata. As software is a crucial piece of the puzzle of emulation, we must also adopt initiatives to preserve software for the long term. Coordinated action is necessary for such an undertaking because it is of interest to all organizations wishing to retain authentic access to digital objects.

A remote emulation service is proposed in order to simplify access to emulated environments. Currently, both the Koninklijke Bibliotheek and the University of Freiburg are involved in creating such a service based on the emulator Dioscuri and GRATE, a specialized remote emulation transport tool. Further refinement of this approach within the Planets Project will result in the next generation of emulation services which will offer centralized access to emulated environments via a generic web interface.

Acknowledgements

Planets is co-funded by the European Union under the Sixth Framework Programme. Planets is a substantial collaborative project that builds on, and brings together, the work of many talented individuals contributing from a consortium of committed organisations.

References

- Reichherzer, T., & Brown, G. (2006). Quantifying software requirements for supporting archived office documents using emulation. *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries, International Conference on Digital Libraries (ICDL)*. USA: Chapel Hill.
- Van der Hoeven, J.R., Lohman, B., & Verdegem, R. (2008). Requirements for applying emulation as a preservation strategy. *The Archiving 2008 Final Program and Proceedings, IS&T Archiving Conference*. Switzerland: Bern.

Van Diessen, R.J. (2002). *Preservation requirements in a deposit system*. Technical Report, IBM / KB Long-term Preservation Study.

Verdegem, R., & Van der Hoeven, J.R. (2006). Emulation: To be or not to be. *The Archiving 2006 Final Program and Proceedings. IS&T Archiving Conference*. Canada: Ottawa.