# Emulation: From Digital Artefact to Remotely Rendered Environments

## Dirk von Suchodoletz, Jeffrey van der Hoeven

University of Freiburg, Koninklijke Bibliotheek
Fahnenbergplatz, 79085 Freiburg, Germany
Prins Willem-Alexanderhof 5, 2509 LK Den Haag, The Netherlands
dsuchod@uni-freiburg.de, jeffrey.vanderhoeven@kb.nl

### Abstract

Emulation used as a long-term preservation strategy offers the possibility to keep digital objects in their original condition and experience them within their original computer environment. However, having only an emulator in place is not enough. To apply emulation as a fully-fledged strategy, an automated and user-friendly approach is required. This can not be done without knowledge of the original software and contextual information about it.

This paper combines the existing concept of a *view path*, which captures contextual information of software, together with new insights to improve the concept with extra metadata. It provides regularly updated instructions for archival management to preserve and access its artifacts. The view path model requires extensions of the metadata set of primary object of interest and depends on additionally stored secondary objects for environment recreation like applications or operating systems.

This paper also addresses a strategy to render digital objects by running emulation processes on distance. The advantages of this strategy are that it improves user convenience while maximizing emulation capabilities.

## 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 advance hardware and software becomes obsolete leaving it uncertain if we still can render today's digital objects in the future. Permanent access to archived digital artifacts thus raises challenges to archive operators who have to deal with keeping access to digital material without loss of information.

Several solutions for long-term access exist of which migration and emulation are the main flavors. Migration – the mostly used digital archiving strategy today – tries to address this problem by changing the digital object to prepare it for access and rendering in future digital

environments. Although this strategy is applicable for 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, solely relying on migration as 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 is relying on a computer environment as well. 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 and Van der Hoeven 2006). For example, the Koninklijke Bibliotheek (KB) and Nationaal Archief of the Netherlands developed Dioscuri (Dioscuri 2008), an x86 emulator developed with the purpose for long-term archiving kept in mind (Van der Hoeven, Lohman, 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 understand the software.

## Bridging the Past to the Future

No matter which emulator is chosen, contextual information of the computer environment is always required. For example, questions such as "for which operating systems is WordPerfect 5.1 compatible?" are less obvious today then twenty 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.

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 be created as well. 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 for 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 whereas in turn the OS relies on particular hardware.
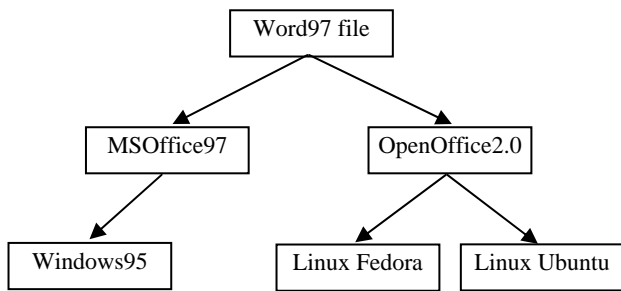


*Figure 1: example view paths*

As dependencies might change in the future, once derived view paths may change over time as well. This is 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).
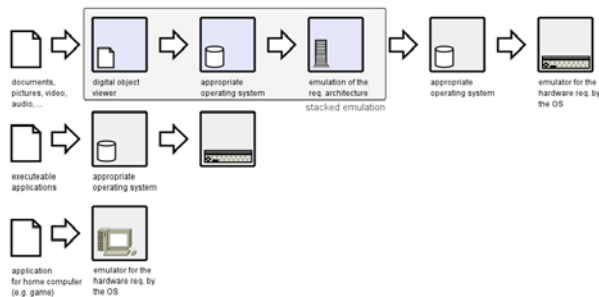


*Figure 2: emulation and view paths*

Looking into the future, the following situations regarding object dependencies can occur:

- 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 discussion as there is only one way to retain access to the digital object. The second situation needs some additional processing. Apparently, 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 decided. To manage various rendering options a procedure will be needed to find the best or 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 needs to be extended to capture metric information. Having applied weightings to all view paths, a classification can be made by what the metric stands for. In general, using view path metrics offer the following possibilities (see also figure 3):

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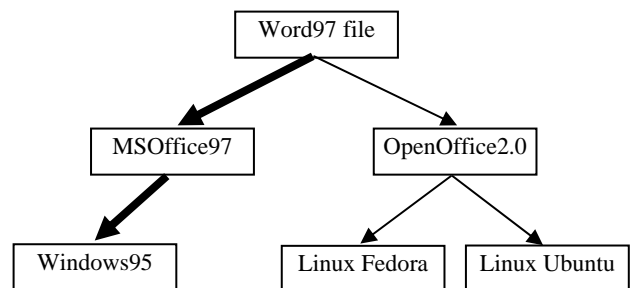


*Figure 3: example view paths with weighting denoted by arrows*

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. Also, weights could be altered to influence a requested rendering in a certain direction. Furthermore, users of computer environments can help evaluating 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 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 affects 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 resulting in various updates in hardware, software and configurations.

Maintenance of each view path and environment brings certain costs with it. Attaching metrics representing operational costs to each view path can deliver an estimate for the effort to spent 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 suggest on formats or prefer specific types of objects over others.

If multiple view paths exist for a given object type, costs would offer another metric to decide on good alternatives. For example, assume a digital object is formatted according the PDF 1.0 Standard and 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 costs for preserving the necessary software in an archive.

## Digital Archive Management

In perspective of emulation, 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.

On ingest, identification and characterisation of digital objects have to be performed. Several solutions exist already of which the most prominent at the moment are PRONOM and DROID of The National Archives in the UK (The National Archives, 2008). However, as these tools are able to offer information about the digital format and some of its dependencies, they do not 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 have 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 might be rejected at ingest time because no guarantee can be given that it will remain accessible over the long term.

Having an emulator and contextual information contained in a view path still leaves some implications at the time a digital object is disseminated and needs to be rendered. Firstly, the original environment consisting of software and hardware needs to be preserved. Secondly, an emulation service is needed to reconstruct the original environment, configure the emulator and activate the emulation process. In the next sections, these two topics will be discussed in more detail.

## 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 authentically. However, emulators only solve one part of the equation. Additional software such as operating system and applications are needed as well (Reichherzer and Brown 2006).

Currently, no standardized or coordinated approach for software preservation exists. Some national libraries treat software releases the same way as publications and preserve them on the shelf next to their books and journals (BnF 2008). Although these software are indexed and managed, the actual bits are still on their original media carriers and are not directly accessible for 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 take care of preserving released software.

To better understand why this area is not yet covered, several reasons can be given that obstruct preservation and access to software. Firstly, the newer the computer environment is, the higher the level of complexity and 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 an application implicitly means all depending sources need 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 still exist that might forbid future generations to use the software. To complicate matters, some software require an online activation or regularly updates to remain operational. Having the software package itself does not work 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 is required to cope 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.

An final interesting challenge 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 with some specific characters, dimensions or the sizes, format of date, calculations and the number of public or religious holidays.

Aside from these obstructions, keeping digital objects alive via emulation the original software needs to be preserved as well. For safekeeping emulators, operating systems, applications and utilities similar guidelines as for digital objects can be applied. 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. Often requested view paths could be stored as combined caches of applications, operating systems and the emulator for faster access. Such specifically prepared containers could be distributed between memory institutions to share the load of management overhead and costs.

## Remote access to emulated environments

Assuming that the required software and metadata is available for emulation, the environment to be emulated has to be prepared. 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 internet. This eases the complex procedures to run emulators and reduces the system requirements of the user to a viewer, preferably a web browser. The user gets the results presented via a virtual screen remotely on its computer. In overview, this kind of setup would offer the following benefits:

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

Still, knowledge about old computer environments is needed to work with emulated computers, but on-screen instructions might offer an extra aid.

Within the Planets project, a pilot is being carried out by 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 which stands for Global Remote Access to Emulation Services and is currently under development by the University of Freiburg. With GRATE any user can easily access emulated environments on distance via their web browser.

First experiments prove that this solution is very user friendly and flexible in configuration. Figure 4 and 5 show two screenshots of GRATE. The first one runs Dioscuri on distance loading WordPerfect 5.1. The second image shows the desktop of Windows98 executed by QEMU emulator (QEMU 2008). Both emulated environments are accessed via a normal web browser.
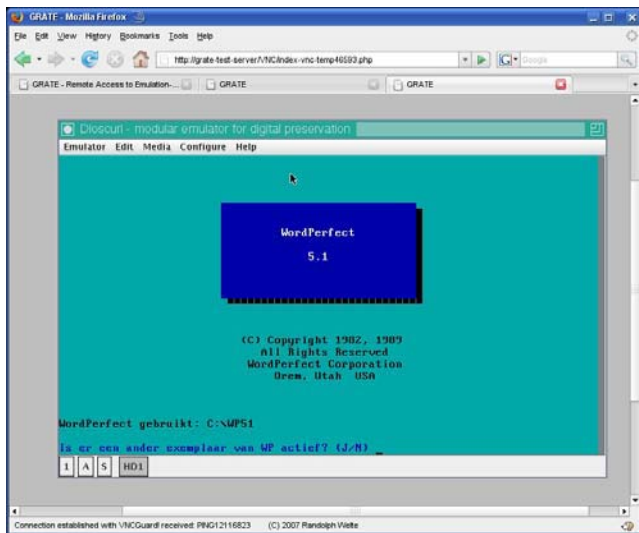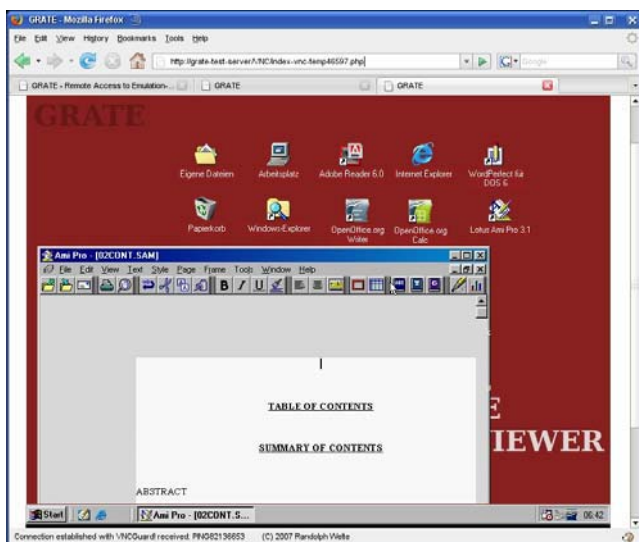
*Figure 4: GRATE running Dioscuri with WordPerfect5.1*



*Figure 5: GRATE running QEMU with Windows98*

The next step is to integrate this emulation service with the interoperability framework of Planets. This will result in a major extension in functionality for preservation action strategies allowing a Planets-user to automatically start emulation activities 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 for successful rendering of the object. Specific information about the object's dependencies on hardware and software should be preserved. Furthermore, to recreate an old computer environment, access to the original software is needed and a access mechanism is required for configuring the emulator and environment.

A flexible solution to manage metadata of environmental dependencies is by using the Preservation Layer Model (PLM). The PLM introduces view paths for each combination of hardware, software and digital file format or collection of files. These formalizations can aid the archivists to manage their digital collections and give them guidelines what to do on object ingest, during object storage and on dissemination. However, the current PLM structure does not explain what to do when multiple view paths can be applied. To overcome this, view path metrics could help to optimize this operation by attaching weights to each layer of dependencies. These weights can be influenced by several reasons such as "most common used operating system" or "user preferred application". The community could even have a vote in the selection procedure by offering feedback and ratings. Moreover, a cost/benefit analysis can be applied to drop less effective view paths.

Aside from metadata the actual software is needed. As software is a crucial piece of the puzzle for emulation, initiatives have to be taken to start preserving software for the long term. This is a task that requires a coordinated action because it is of interest for all organizations that would like to retain authentic access to digital objects.

To simplify access to emulated environments, a remote emulation service is proposed. 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, offering centralized access to emulated environments via a generic web interface.

## References

Verdegem, R., and Van der Hoeven, J.R., Emulation: To be or not to be. In proceedings of the Archiving 2006 Final Program and Proceedings, 55 – 60. Ottawa, Canada: IS&T Archiving Conference.

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

Van der Hoeven, J.R., Lohman, B., Verdegem, R., Requirements for Applying Emulation as a Preservation Strategy. In proceedings of the Archiving 2008 Final Program and Proceedings. Bern, Switzerland: IS&T Archiving Conference.

Van Diessen, R.J., 2002. Preservation Requirements in a Deposit System, Technical Report, IBM / KB Long-term Preservation Study.

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

Reichherzer, T., Brown, G., Quantifying software requirements for supporting archived office documents using emulation. In proceedings of Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries, 86 – 94. Chapel Hill, USA, International Conference on Digital Libraries (ICDL) 2006.

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

QEMU website, http://bellard.org/qemu/

## Acknowledgements