

The International Journal of Digital Curation

Volume 8, Issue 1 | 2013

Towards Emulation-as-a-Service: Cloud Services for Versatile Digital Object Access

Dirk von Suchodoletz, Klaus Rechert and Isgandar Valizada,
University of Freiburg

Abstract

The changing world of IT services opens the chance to more tightly integrate digital long-term preservation into systems, both for commercial and end users. The emergence of cloud offerings re-centralizes services, and end users interact with them remotely through standardized (web-)client applications on their various devices. This offers the chance to use partially the same concepts and methods to access obsolete computer environments and allows for more sustainable business processes. In order to provide a large variety of user-friendly remote emulation services, especially in combination with authentic performance and user experience, a distributed system model and architecture is required, suitable to run as a cloud service, allowing for the specialization both of memory institutions and third party service providers.

The shift of the usually non-trivial task of the emulation of obsolete software environments from the end user to specialized providers can help to simplify digital preservation and access strategies. Besides offering their users better access to their holdings, libraries and archives may gain new business opportunities to offer services to a third party, such as businesses requiring authentic reproduction of digital objects and processes for legal reasons. This paper discusses cloud concepts as the next logical step for accessing original digital material. Emulation-as-a-Service (EaaS) fills the gap between the successful demonstration of emulation strategies as a long term access strategy and its perceived availability and usability. EaaS can build upon the ground of research and prototypical implementations of previous projects, and reuse well established remote access technology.

In this article we develop requirements and a system model, suitable for a distributed environment. We will discuss the building blocks of the core services as well as requirements regarding access management. Finally, we will try to present a business model and estimate costs to implement and run such a service. The implementations of EaaS will influence future preservation planning in memory institutions, as it shifts the focus on object access workflows.



Introduction

The IT world is virtualizing, and more and more services get abstracted from the machine (e.g. computer) a user is interacting with. This is supported by established remote control and interaction protocols, as well as a rising number of different cloud services. Networked applications, such as browser-based software, is becoming more commonplace. Cloud services, both storage clouds like iCloud, Dropbox or Azure as well as services and systems like Google Docs, various web mail services or remote online gaming,¹ are in use pretty often already. These offerings are marketed as the next logical step in systems architecture, superseding the seasoned client/server model dating back to the 1980s. Together with often ubiquitous network connectivity, these developments allow businesses and institutions to offer services for a wide range of mobile and other types of performance-restricted devices.

Implementation of successful cloud services is challenging, but in successful cases, organisations can save substantial costs and are able to adapt the organisations' business processes to its customers' needs. The application of cloud services is not necessarily limited to today's systems. The ongoing developments in clouds and the research in emulation can be combined into a future preservation strategy offering access to a wide range of digital objects, as well as preserved applications and systems. The most important technology components in emulation-based preservation strategies are obviously emulators. While the technical challenges involved in developing emulators are not considered in this paper, usability and accessibility of emulators for non-technical users are crucial. Emulation technology usually resembles a specific computer system. Since the number of different ancient and current computer systems is limited, the number of required emulator-setups is limited too. Hence, providing access to an emulation is suitable for standardized services. A cloud-based strategy in emulation is able to efficiently reduce the effort individuals or organisations have to invest in order to keep their digital artefacts accessible. An emulation-backed strategy will focus on the long-term availability of original environments to offer access to a wide range of object types, without the need to migrate them.

Preservation steps necessary for a specific platform have to be repeated for other platforms too. Furthermore, knowledge of ancient software environments diminishes. Hence, the amount of effort increases with the rising number of deprecated (hardware) platforms, as emulation is still a complex and often labour-intensive task. Thus, increasingly single organisations or persons are swamped by the challenge of reproducing original environments from a large number of software packages and components. Beside the technical obstacles, there exist a couple of legal concerns regarding proprietary firmware and software licenses to run original environments within hardware emulators.

Optimally, instead of installing a huge number of software packages that are difficult to maintain even for a small number of relevant platforms, the user should simply be able to install a simple access application made available for a wide range of today's and future platforms. We suggest remote access to a new type of cloud

¹ Remote system access services, such as Gaikai (<http://www.gaikai.com>) or OnLive (<http://www.onlive.com>).

service – Emulation-as-a-Service (EaaS) – as a possible solution. Remote access applications and protocols are to be defined to abstract from and translate the actual capabilities of the chosen local platform to the remote running service interfaces. This is not a particular challenge, as the same base principles are valid for accessing recent and obsolete environments over computer networks. Emulated original environments could then “blend” in seamlessly with actual services. These considerations lead to a solution that allows access to a 1985 home computer a game running in MESS,² access to mid-2000 Linux, Windows or Sun Solaris desktops, access to a Macintosh PowerPC of the end-1990s and some modern 3D games through the same application, representing a front-end interface to the emulation services. The separation of the service from the user interface allows a distributed environment. Services like emulation components, software archives and authentication services can be shared and split among several institutions and third-party providers to enable specialization following the division of labour principle.

For the end user not much needs to be changed: digital preservation (DP) services are delivered using established technology. This would allow similar models for service accounting and access regulation as for existing services. Cost models are easier to deploy and intellectual property rights (IPR) issues are less difficult to solve. Plus, a range of different paid access services to preserved material could be established by the content holders or third-party service providers.

Related Work

Emulation-as-a-Service is not an entirely new concept. It is build on past and on-going research projects, as well as on well established technologies. The PLANETS project (Schmidt et al., 2010; King et al., 2009) introduced the concept of interconnected (web-)services binding different digital preservation workflows into a single framework. These services do not necessarily need to run locally, but can be deployed in a distributed manner and can be accessed remotely. During the PLANETS project we developed the prototype GRATE,³ which allows the wrapping of various software environments within a single networked application. Designed as a general purpose remote access system to emulation services, the architecture provides users with an abstract interface independent of the digital object’s type and was linked to other web-services like PLATO (Becker et al., 2009). The prototype showed a couple of shortcomings, including static-only original environments and very restricted object transport and loading capabilities. GRATE-R became the next iteration, changing the approach to remote emulation regarding the user interface. Instead of exporting a complete host system GUI with an emulator running, just the emulator screen output and mouse and keyboard inputs are exposed to the remote client (von Suchodoletz and Cochrane, 2011).

KEEP (Schmidt et al., 2010) deepened the understanding of emulation and especially on aspects like integration and usable frameworks. The EU-sponsored

² Multi Emulator Super System (<http://mess.org>) is an open source project which documents the hardware for a wide variety of (mostly vintage) computers, video game consoles, and calculators through software emulation.

³ Global Remote Access To Emulation. J. van der Hoeven of the Dutch National Library first coined the term Emulation-as-a-Service for it at the Missing Link WA conference 2009 in London: http://www.dpconline.org/component/docman/doc_download/400-0907vanderhoevenmissinglinks


SCAPE is an ongoing research project with a focus on the scalability of planning and execution of institutional preservation strategies (Conway, Lambert and Matthews, 2011). The project develops infrastructure and tools for scalable preservation actions, provides a framework for automated, quality-assured preservation workflows and supplements these components with a policy-based preservation planning and watch system (Becker and Rauber, 2011; Schmidt and Rella, 2011). The current TIMBUS project researches into resilient business processes and considers their execution context. Business processes should be made available over long periods through a set of activities carried out in the isolation of a single domain. The project considers the dependencies on third-party services, information and capabilities that will be necessary to validate digital information in a future usage context (Barateiro et al., 2012).

With respect to emulation services, usability, access and scalability have not been well addressed yet. While the principle techniques and methods are well-researched, standardized interfaces, workflows and best practice guidelines for emulation processes in digital preservation are still at a very early stage. Emulation-as-a-Service is able contribute to these aspects by lowering the hurdles to experiment with this technology, especially for practitioners.

Motivation for a Cloud-Based Service

With the KEEP emulation framework (Lohman et al., 2011) aiming at a wider range of today's platforms, it became clear that it is challenging to provide various hardware emulators and framework services on very different hardware for several, mostly technical, reasons. Architectural and technical differences between powerful desktop machines, notebooks, thin clients, mobile devices or even next generation TV screens are significant and thus difficult to be bridged with a single solution. Additionally, often secondary software components (e.g. operation systems, ROMs, firmwares, drivers, etc.) are required at the user's site. A similar problem arises from access-restricted digital artefacts. If a user is interested in such an object, a memory institution may not be able to hand over the complete object due to legal or privacy issues. By using a cloud service model with remote emulation, some of these problems could be tackled quite efficiently:

1. Development and maintenance of emulators and their associated DP frameworks could be focused on only a few current architectures, leading to a controlled and well understood environment. This avoids the complexities of cross-platform development and allows easier testing, as fewer targets with fewer variety are to be considered.
2. Original environments and their various software components, as well as the requested artefacts, are not leaving the controlled systems of service providers. Thus, EaaS can offer users of memory institutions the various contents from their digital collections, such as digital art, encyclopedias, primary scientific data, teaching materials or a famous person's original working environment (Loftus, 2010; Paul-Choudhury, 2011) without giving them direct access to the primary data itself.

- 
3. Use of standardized and well established remote access applications and protocols allow platform independence. Specifically, the applications also have to adapt to different input/output methods.
 4. The access to certain digital artefacts or complete environments can be controlled in a more effective fashion, preventing the user from copying material or analysing it in an unwanted way. Thus, IPR issues are mitigated, as no original artefacts get copied over to the end user's machine. The user is able to deal with the object as in a controlled lab environment. Certain actions can be restricted by filtering or suppressing inputs and certain outputs.

Cloud services are usually offered in different flavours and not every variant necessarily matches the organization's needs. The "cloud" is usually defined as an architecture that allows an individual or organization to use tools, platforms and infrastructures over a computer network remotely, and to incorporate them into their own business processes. The literature typically defines three distinguished levels of cloud technology (Vaquero et al., [2009](#)):

- **Software as a Service (SaaS):** Using software tools or even complete software suites remotely.
- **Platform as a Service (PaaS):** Using predefined remote computer platforms running specific system software, such as databases or software development environments.
- **Infrastructure as a Service (IaaS):** Using storage/computing elsewhere.

Emulation fits into the model on different layers. A potential business model could be based on a PaaS model, allowing users to re-run an ancient computer platform. In such a setting EaaS is not limited to single machines, but allows users to run complete networked applications in a well defined, secured (virtual) network environment.

While EaaS does not primarily require large computing power or storage capacity, the model very well benefits from cloud technology in terms of remote access, distribution of services, established authentication and accounting frameworks. The EaaS model can help to leverage typical cloud advantages for emulation processes in DP: scalable, on-demand services, less waste of compute resources, optimize costs or solve IPR related challenges. Plus, the cloud creates a certain pressure for the standardization of services. All together, EaaS offers the chance for better user-centered access services in DP. Resources can be scaled to the actual needs of the organization and thus provides great flexibility, e.g. pay-per-use business models or targeted short-term contracts.

Emulation-as-a-Service

Memory institutions can decide upon their organizational policies if they deploy a private or use a public cloud, either providing services in-house or use/provide third party offerings. In the first case, the operation services are close to the objects to be accessed. Thus, this approach provides a high degree of availability, security and confidentiality. As private clouds are in full control of the hosting institution they are

able to deal with proprietary or restricted environments, such as preserved research systems, product or software development, or sensitive business processes. For instance, passing digital artefacts over to third parties is not required, and therefore direct enforcement of privacy policies is possible.

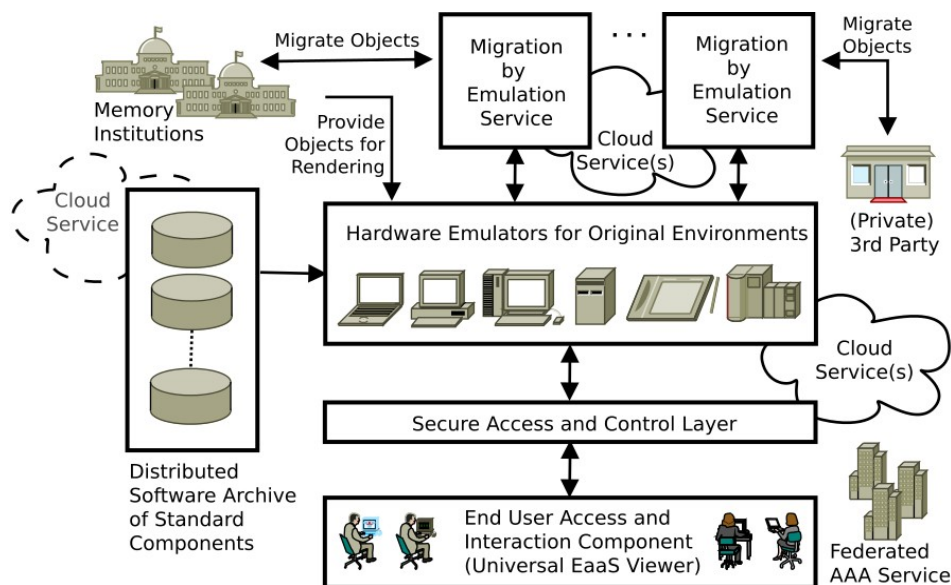


Figure 1. Components and service modules of EaaS.

In contrast, by using a public cloud, institutions do not need to maintain their own installations and can avoid underutilized, costly, special purpose server parks. Public clouds are suitable for artefacts that are meant to be openly available, such as online collections of computer games, the presentation of digital art or general access to public data of various kinds.⁴ Besides providing access to deprecated computer systems, new types of services might be established. Running systems can be frozen and resumed by different users, or offered to be run from a certain execution state. Furthermore, parallel access of several users to the same system is possible, e.g. for performance measures, scientific, guidance or teaching purposes.

Centralization

EaaS as a centrally accessible, distributed service is able to help memory institutions with the de-duplication of efforts required to re-enact complete original software environments, such as the storage of software components. It becomes less costly to ensure redundancy of installations and running services. The management of a software archive containing both the core components, such as emulators and components for the (re-)production of original environments, as well as prepared environments for caching often required objects, can be simplified through EaaS.

Specialization

Networked services support different usage models, such as using an EaaS service as a single workflow component or as part of a complete workflow, and thus offer

⁴ Following the open data initiatives of the various governmental entities.

institutions the ability to specialize on their core competencies. Memory institutions can specialize and concentrate their knowledge on certain platforms. Cross-institutional cooperation would complement the local services.

Scalability

Since the number of different ancient and current computer systems is limited, the number of required emulator setups is limited too. Preservation planning and preservation costs are fixed, determined only by the number of emulators and emulated systems. Thus, such an emulation-based approach scales perfectly with the number of the primary digital objects.

Emulators as Service Backends

Emulators are the technical key component in EaaS. Depending on the original environments to be accessed or rendered, a certain number of different hardware emulators are required. They provide both the base layer for the original system environment and several interfaces for user interaction, handling and framework integration (von Suchodoletz and Cochrane, [2011](#); Rechert et al., [2012](#)). They bridge the gap between outdated computer platforms and actual software and hardware.

As the original environments are meant to be available to users on their actual devices, access components to emulators also need to translate machine in- and output concepts. Not only screen resolution and color depth have increased over time, but also keyboard layouts have changed and different types of inputs, such as a wide range of mice, joysticks or recently, position sensors, have been added. While a cloud approach to emulation simplifies a number of issues, several challenges of the emulation strategy still need to be solved:

- Appropriate hardware emulators need to be available to support the original environments running in the host system of the cloud service.
- The cloud service provider has to have access to required original software applications, operating systems, firmwares and drivers, including the appropriate rights to use them.
- The ability and knowledge to re-run the original software environment is required, which means users will need to know how to operate the emulator, install and load the complete software stack, and execute it.
- Emulator and original environment-specific workflows are required to prepare and transport the original artefacts to be used in and extracted from the emulated environments.

However, a cloud solution allows the provisioning of emulators to take place in well-controlled environment, which is easier to define and to maintain compared to end user systems. It focuses the available resources and allows the specialization and division of labour among the involved institutions.



Linking Back and Front Ends

Different emulation services for original environments should be made accessible in a unified fashion to avoid duplicated efforts in client programming and to provide a consistent user interface. All computer platforms have similar characteristics, such as screen output, audio, different methods for input, and can get switched on/off. Some of them are able to accept various (removable) media for data exchange and booting. This functionality needs to be abstracted from the concrete emulator or virtual machine and made available remotely. A couple of concepts and implementations for local access like VMware vCenter, RedHat Virtual Machine Manager⁵ or the KEEP Emulation Framework (Lohman et al., [2011](#)) can provide ideas for client side requirements.

For general remote screen access, a couple of well-established proprietary and open protocols exist from the simple VNC/RFB to more powerful RDP or Citrix. Low latency requirements and 3D transport was tackled by streaming services like OnLive or Gaikai. A generic client interface API needs to be defined, which can be easily implemented for a wide range of host platforms and can connect to a wide range of target hardware emulators. This API needs to be extensible over the time, but should focus to match to the capabilities of power limited mobile devices.

As computer systems change in many aspects over the time, the API needs to be able to translate and to emulate the several forms of in- and outputs. This could require the addition of, for example, an overlay keyboard or game console controller, or the provision of a virtual mouse on touch screen devices. The screen resolution might need to be adapted as well. While, for instance, the home computers of the 1980s produced screen resolutions for TV sets, today's devices offer a significantly higher resolutions and the resulting images should sufficiently fill the screen (Figure 1). Alternatively, fusion-like⁶ interfaces might be desirable. Plus, an overlay help functionality might be implemented into the screen output layer.

To provide remote user interaction for EaaS, emulators need to expose appropriate interfaces. Some emulators and virtual machines have built-in VNC or RDP. Alternatively, the Simple DirectMedia Layer library (SDL),⁷ which is used by many open source emulators, can be patched to use VNC for video outputs. The VNC approach is limited though: it usually does not feature an audio channel.⁸

For future emulator development, the development of a streaming interface with similar characteristics as the aforementioned streaming services should be considered. The interface should handle both video and audio, and should cope with additional forms of input from newer types of input events, for example. To allow easy sharing of common resources, like certain original environments, emulators should implement

⁵ For further information and background, see: <http://virt-manager.org>

⁶ As provided by some virtual machines being able to blend in single applications from the original environment into the actual desktop.

⁷ SDL (<http://www.libsdl.org>), is a cross-platform multimedia library designed to provide fast access to the graphics frame buffer and audio device.

⁸ This is a special challenge, as audio and video take pretty much different "paths" (system-wise) until they reach the consumer and there is no a trivial way of re-syncing the video and audio streams.

copy-on-write disk access: this would allow many instances to use exactly the same copy to serve different users at the same time.

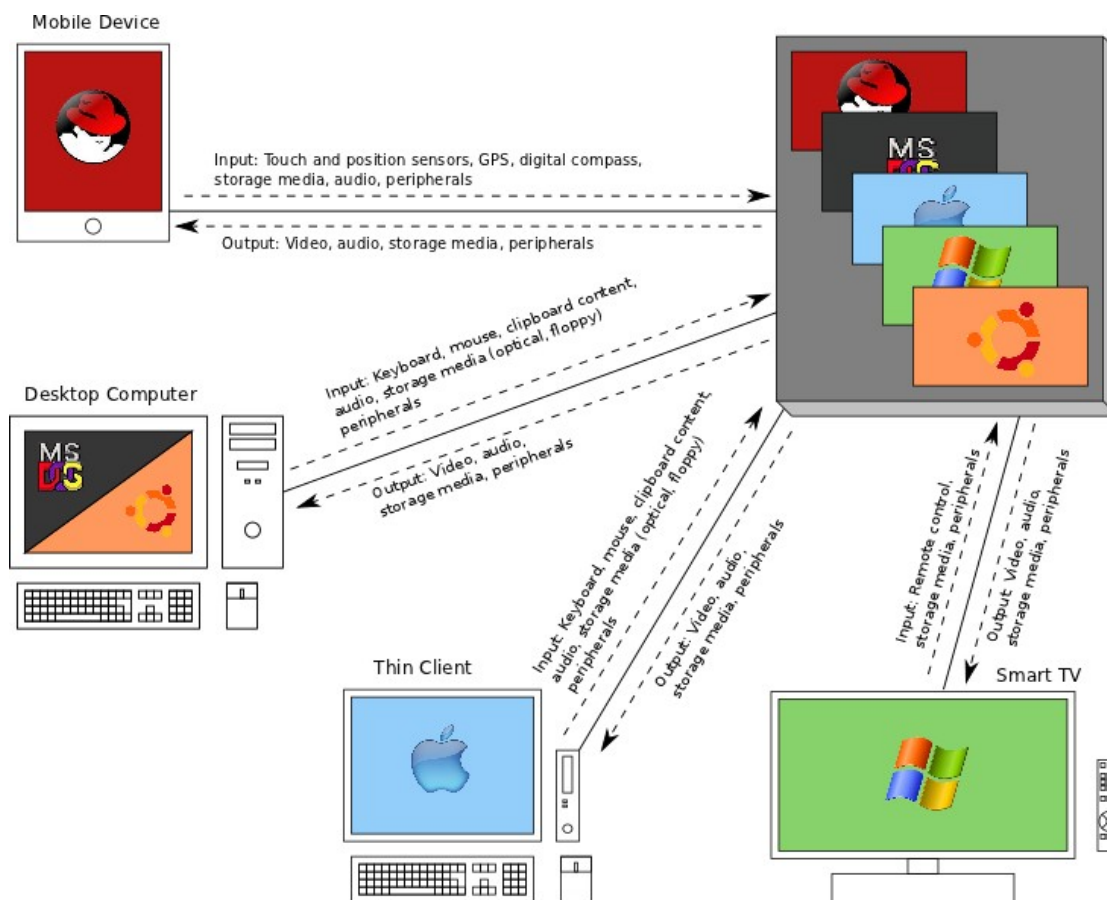



Figure 2. The remote access protocol needs to transport and translate different types of in- and outputs.

Most of the traditional in- and outputs can be rather straightforwardly mapped or emulated, and the border between the remote emulation back-end and user front-end is pretty clear. It will be more difficult in special cases if, for example, a local joystick should be used or a hardware dongle needs to be attached. For such hardware items a local emulation component which produces the proper translations for the back-end will be required.

The existing set of tools and protocols is not completely suitable yet. Many virtual machine managers use VNC and provide additional channels for machine remote control, but lack features like audio, or machine controls like power, reset or resume. Plus, 3D is still a challenge for most of the existing protocols. Developments like HTML5 ease the development of web-clients and thus would help to abstract from client programming for different target platforms. The HTML5 standard is a major step towards the standardization of streaming services for a wide range of clients and devices. Together with proxy tools like Guacamole,⁹ an open source project to provide an abstract interface to remote desktop protocols like VNC and RDP, the screen output and user input can be realized on nearly every modern device. At the moment

⁹ See: <http://guac-dev.org>



the existing remote access protocols do not completely satisfy the remote emulation requirements, as additional data channels for audio, remote block device access or event recording back channels are lacking. A new, unified protocol can both improve the security and privacy of remote access, as well as simplifying the implementation for a wide range of end user clients.

Business Model

Emulation-as-a-Service offers several advantages. It allows new stakeholders to enter the market, as services can be offered to a wide range of different customers remotely. Memory institutions can use their knowledge and advantage in the field of digital preservation and access to provide paid services to commercial entities, which seek to solve their preservation needs and conform to legal requirements. EaaS can profit from the economics of the long tail, as a rather special service can be provided virtually to the whole world. It can be run as a distributed service and thus allow specialization between different memory institutions or third party service providers. Plus, a distributed approach allows the economics of the long tail. There exist well-established cost and payment models, such as pay-per-view or flat rate arrangements, which can be applied. This allows the re-use of existing business models and can generally help with more transparent cost calculations. Nevertheless, the general cloud maintenance costs are to be weighted against preservation – and access-specific costs.

When considering cloud technology a distinction is to be made between (large) organizations that run their own infrastructure and (small) organizations that do not or are considering whether to develop one. For (small) organizations that do not have their own scalable infrastructure, cloud technology can be an attractive (partial) solution. Depending on the type of digital objects and original environments treated within an organization, preservation and access services might be run in the private cloud only. However, the same organization might offer validation services to third party. EaaS can help to share the costs between institutions, offer regional distribution of efforts and cooperation, and may avoid vendor lock-ins. EaaS supports the sharing of costs for efforts like the maintenance of emulators and the software archive of required components.

Conclusion

EaaS offers the chance to solve several challenges of emulation strategies in DP. The complex part of the implementation is kept in memory institutions or handled by third party providers. EaaS can provide a seamless, long-term perspective for systems-as-a-service and use the same technology to access today's and past systems. Optimally, there is no need for the user to deploy a DP-specific access client. The cloud acts as a base technology for cooperation, as it helps to standardize and to improve the service to be acceptable for third party users, such as other memory institutions. In a distributed EaaS model, the costs of archiving secondary digital objects can be shared. With mutual specialization, niches and specific areas can be covered without losing generality. Emulation and emulators will become more easy to handle. They do not need to be adapted permanently to an ever changing technological landscape of end user's devices, but can be maintained for the limited

←—————→

set of cloud backend platforms. The efforts to adapt between future and past technologies can be shifted to the access protocol, mitigating the pressure to continually update the emulators.

Similarly, for potential users requiring guaranteed long-term access to their digital objects, creation of the aforementioned EaaS is sufficient. If institutionalized archives guarantee the long-term availability of the identified component within their network, no additional archival cost occur. In combination with a distributed access infrastructure structure, emulation-as-a-service, preservation planning and preservation costs are fixed, determined only by the number of emulators and emulated systems. Thus, this emulation-based approach scales perfectly with the number of primary digital objects. The costs of the archival network, as well as the cost of curation, will be reduced by every new user and object.

EaaS also mitigates several IPR challenges. No digital artefact or secondary software component needs to leave the institution providing the service. Delivered objects can be watermarked, lowered in the rendering quality or otherwise restricted to preserve privacy issues and conform to legal requirements. Thus, it gets easier for both sides, with providers offering access to a wide range of digital artefacts and users being able to consume them.

The bwFLA (Rechert et al., [2012](#)) project, a functional long-term archive, has started implementing and integrating emulation-as-a-service as part of an state-wide initiative. Currently, the bwFLA EaaS supports eight different emulators and is able to run 15 distinct legacy computer platforms. The platforms range from MacOS 7 running on a MK68 system emulator, PPC-based platforms to various x86-based platforms. Each emulation component is available to be used in various archival workflows through a common web service interface.

Acknowledgements

Parts of the work presented in this publication have been researched and implemented in the bwFLA project, sponsored by the federal state of Baden-Württemberg, Germany.

References

- Barateiro, J., Draws, D., Neumann, M.A. & Strodl, S. (2012). Digital preservation challenges on software life cycle. Paper presented at the Conference on Software Maintenance and Reengineering (CSMR). Szeged, Hungary.
- Becker, C., Kulovits, H., Kraxner, M., Gottardi, R., Rauber, A. & Welte, R. (2009). Adding quality-awareness to evaluate migration web-services and remote emulation for digital preservation. Paper presented at the 13th European Conference on Digital Libraries (ECDL09). Corfu, Greece.
- Becker, C. & Rauber, A. (2011). Decision criteria in digital preservation: What to measure and how. *Journal of the American Society for Information Science and Technology*, 62, 1009–1028. [doi:10.1002/asi.21527](https://doi.org/10.1002/asi.21527)

- Conway, E., Lambert, S. & Matthews, B. (2011). Managing preservation networks: Issues of scale for scientific research assets. In *Proceedings of the 8th International Conference on Preservation of Digital Objects (iPRES2011)*. National Library Board Singapore & Nanyang Technology University.
- King, R., Schmidt, R., Jackson, A.N., Wilson, C. & Steeg, F. (2009). The planets interoperability framework. Paper presented at the 13th European Conference on Digital Libraries (ECDL09). Corfu, Greece.
- Loftus, M.J. (2010). The author's desktop. *Emory Magazine*, 85(4).
- Lohman, B., Kiers, B., Michel, D. & van der Hoeven, J. (2011). Emulation as a business solution: The emulation framework. In *Proceedings of the 8th International Conference on Preservation of Digital Objects (iPRES2011)*. National Library Board Singapore & Nanyang Technology University.
- Paul-Choudhury, S. (2011). Digital legacy: Respecting the digital dead. *New Scientist Online*. Retrieved from <http://www.newscientist.com/article/dn20445-digital-legacy-respecting-the-digital-dead.html>
- Rechert, K., Valizada, I. & von Suchodoletz, D. (2012). Future-proof preservation of complex software environments. In *Proceedings of the 9th International Conference on Preservation of Digital Objects (iPRES2012)*. Retrieved from <https://ipres.ischool.utoronto.ca/sites/ipres.ischool.utoronto.ca/files/iPres%202012%20Conference%20Proceedings%20Final.pdf>
- Rechert, K., Valizada, I., von Suchodoletz, D. & Latocha, J. (2012). bwFLA: A functional approach to digital preservation. *PIK*, 35(4), 259–267. [doi:10.1515/pik-2012-0044](https://doi.org/10.1515/pik-2012-0044)
- Schmidt, R., King, R., Jackson, A., Wilson, C., Steeg, F. & Melms, P. (2010). A framework for distributed preservation workflows. *International Journal of Digital Curation*, 5(1), 205-217. [doi:10.2218/ijdc.v5i1.154](https://doi.org/10.2218/ijdc.v5i1.154)
- Schmidt, R. & Rella, M. (2011). *An approach for processing large and non-uniform media objects on MapReduce-based clusters*. Lecture Notes in Computer Science, 7008, 172–181. [doi:10.1007/978-3-642-24826-9_23](https://doi.org/10.1007/978-3-642-24826-9_23)
- von Suchodoletz, D. & Cochrane, E. (2011). Replicating installed application and information environments onto emulated or virtualized hardware. In *Proceedings of the 8th International Conference on Preservation of Digital Objects (iPRES 2011)*. National Library Board Singapore & Nanyang Technology University.
- Vaquero, L.M., Rodero-Merino, L., Caceres, J. & Lindner, M. (2009). A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39(1), 50-55. [doi:10.1145/1496091.1496100](https://doi.org/10.1145/1496091.1496100)