Demo: Migration-by-Emulation

Isgandar Valizada
Albert-Ludwigs University
Freiburg, Germany
isgandar.valizada@rz.uni-freiburg.de

Klaus Rechert
Albert-Ludwigs University
Freiburg, Germany
klaus.rechert@rz.uni-freiburg.de

Dirk von Suchodoletz
Albert-Ludwigs University
Freiburg, Germany
dirk.von.suchodoletz@rz.uni-freiburg.de

ABSTRACT

The availability of migration tools for older formats is often limited. Thus we suggest a different approach: using the original applications to access the object and transfer the latter into formats which can be accessed in today's environments. The appropriate environment for the digital artefacts could be provided through emulation. With the reproduction of the original environment, a large and diverse set of migration input/output paths becomes available. Working for the Open Planets Project the authors the authors created remotely accessible Web services integrated into the PLANETS testbed. These services demonstrate preservation workflows using migration together with the emulation of original environments.

1. CONCEPT

A strategy for accessing digital artifacts with outdated formats is to convert them into formats which can be rendered or executed in todays digital ecosystems. In most cases the applications or operating systems developed by the software producers are the best candidates for handling them. Emulation is the best way to reproduce original digital environments, which themselves provide the base layer for very flexible multiple migration input/output scenarios. Typically, applications used to produce or render digital objects were programmed with a human user in mind. He operated the application through keyboard or mouse interaction. Many of those applications don't provide programming interfaces for unattended command line operation e.g. to perform a format migration. However, performing migrations manually for every digital object is not a feasible strategy in many cases; because of the large quantities held by many institutions, it may turn out to be a time-consuming and costly task. Additionally, depending on the original environments, many archivists or private users don't have the requisite knowledge on either how to install a certain application or operating system or how to handle a certain emulator.

iPRES2011, Nov. 1–4, 2011, Singapore.

The idea presented in this demo is the creation of easy-to-use migration tools out of a combination of original environments running in emulators and steered by automated UI interaction. In [3] the authors showed the general feasibility of recording and replaying interactive user sessions in an abstract way. We brought the development further by implementing services which conform to the PLANETS interoperability framework *migrate* Web Service interface (cf. [1]). Using this framework complex procedures to transform a digital object into a selected output format [2] could be deployed. In contrast to simple command-line input-output migration tools, a migration-by-emulation service needs a more complex initial setup:

- *System Emulation*: Hardware emulation including a full reconstruction of an ancient environment. For instance a i386 CPU, ISA Systembus, VESA compatible graphics, PS/2 mouse and AT keyboard are minimal requirements, e.g. for Windows 3.11.
- *System Environment*: An appropriate runtime environment (e.g. a disk image file) preconfigured with the operating system, necessary drivers and tools, and the required target application. Furthermore, each environment specifies at least one transportation option, defining how digital objects can be injected into and extracted from the virtual environment. Examples range from different kinds of floppy-disk images to hard-disk container formats and advanced networking options.
- *Interactive Workflow Description*: An abstract description of all interactive commands to be carried out in order to perform a certain migration. Such an description consists of an ordered list of interactive input actions (e.g. key strokes, mouse movements) and expected observable output from the environment (e.g. screen- or system-state) for synchronization purposes.

The created service is split into two parts. The scenario preparation unit (Fig. 1), typically run once, interactively records the user interaction and creates the event list for playback. The playback unit is used by the migration service and re-runs the once recorded events unattended.

2. MIGRATION-BY-EMULATION

As migration-by-emulation services should be accesible the same way as standard command line tools, they are registered and deployed using the same methods within the

PLANETS testbed [4]. The Java-programmed prototype for complex emulation-backed migration workflows has all core components implemented. The services are called from within the testbed standard procedures. Preconfigured original environments are deployed and the Grate-R VNC record service (Fig. 1) is used to generate abstract workflow descriptions. The generated interaction tracefiles were then attached manually to the appropriate original environment to form a migration unit. Two different migration services

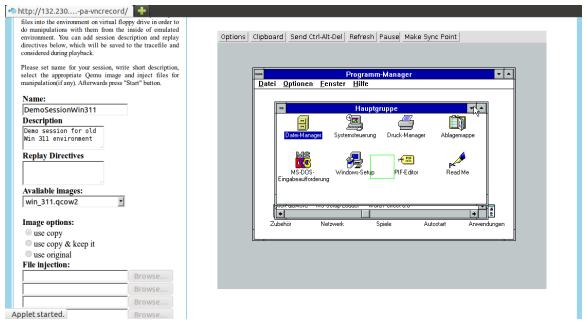

Figure 1: Scenario preparation unit and recording Web front-end

are registered within the testbed. One of them is a truly atomic migration accepting WPD as input and producing RTF as output. Thus the resulting file is directly delivered to the user after the procedure finishes. A migration took less then a minute per item and succeeded on a range of different input files. The second service is more complex as it takes an AMI Pro text document (SAM) as input and produces two different outputs, a TXT and a PDF (Fig. 2). The TXT is the result of a classical "save-as" migration. The PDF is generated by sending the document to a virtual printer generating PS as output. This file is then loaded onto the Ghostview application, which renders a PDF from it. This migration unit was deliberately of a more complex nature. We wanted to demonstrate the feasibility of producing more than one output file from a single input. This helps to evaluate and compare different workflows in regard to runtime, reliability and complexity. Nevertheless, the framework interfaces are to be extended to accommodate more flexible workflows which produce more than one result from a single input.

Additionally, a virtual disk-handling service was programmed to produce disk image containers for different emulators with the option to specify a range of supported filesystems understood by the original system environments. The creation of a QEMU compatible container with a FAT filesystem in it is comparatively simple; other containers and filesystems are supported to by using the "qemu-img" container conversion tool.

3. CONCLUSION

Our implementation focused on the feasibility of the preservation framework integration. Future research is dedicated to the speeding up of workflows by looking into the VNC recording and playback. The tracefiles are a good starting point for optimizaton. They could be enriched with additional metadata to use them for progress reporting. A certain state in the metadata directly correspondends to a state

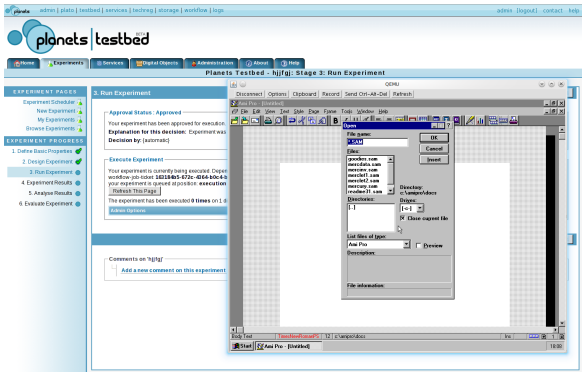

Figure 2: Migration experiment run in Planets testbed. QEMU interface connection opened for control and demonstration purposes.

of the migration workflow and could be reported back to the preservation framework. The tracefiles could be modularized to better identify the different stages, like original operating system booting, application starting, artefact loading, and saving in a new format. This information could be used not only for feedback but also to identify checkpoints. Those checkpoints could help with error recovery for restarting the procedure after failed attempts. Plus, these workflows could help to evaluate future versions of emulators before they get integrated into preservation systems. These issues are part of the ongoing research at Freiburg University.

With the integration of migration-by-emulation into the PLANETS testbed, such migration tasks become available to a wider community and hopefully encourages a range of different institutions to test, create and deploy such services. Since the individual migration services only require the preparation of system environments and the production of appropriate recordings, the proposed system is able to speed up the creation of diverse migration services, since no additional programming or integration effort is required.

4. REFERENCES

[1] Ross King, Rainer Schmidt, Andrew N. Jackson, Carl Wilson, and Fabian Steeg. The planets interoperability framework. In *Proceedings of the 13th European Conference on Digital Libraries (ECDL09)*, pages 425–428, 2009.

[2] Klaus Rechert, Dirk von Suchodoletz, and Randolph Welte. Emulation based services in digital preservation. In *JCDL '10: Proceedings of the 10th annual joint conference on Digital libraries*, pages 365–368, New York, NY, USA, 2010. ACM.

[3] Klaus Rechert, Dirk von Suchodoletz, Randolph Welte, Maurice van den Dobbelsteen, Bill Roberts, Jeffrey van der Hoeven, and Jasper Schroder. Novel workflows for abstract handling of complex interaction processes in digital preservation. In *Proceedings of the Sixth International Conference on Preservation of Digital Objects (iPRES09)*, 2009.

[4] Rainer Schmidt, Ross King, Andrew Jackson, Carl Wilson, Fabian Steeg, and Peter Melms. A Framework for Distributed Preservation Workflows. *International Journal of Digital Curation*, 5(1), 2010.