EXECUTABLE ARCHIVES

Software integrity for data readability and validation of archived studies

Natasa Milic-Frayling

Intact Digital Ltd

United Kingdom

natasamf@intact.digital

0000-0003-1244-1973

Marija Cubric

University of Hertfordshire

United Kingdom

m.cubric@herts.ac.uk

0000-0001-6699-3576

Abstract – This paper presents practices and processes for managing software integrity to support data archiving for long term use in response to the regulatory requirements. Through a case study of a scientific software de-commissioning, we revisit the issues of archived data readability. Established software lifecycle management processes are extended with archiving and data integrity requirements for retention of data and revalidation of data analyses. That includes the software transition from operational to archival use within the Executable Archive model that extends the traditional data archive with computing environments with software installations required to reproduce study results from the archived records. The content use requirements are an integral part of both data access and the software management considerations, assuring that data integrity is fully supported by the software integrity.

Keywords – data integrity, software integrity, study reconstruction, significant properties, executable archive

Conference Topics – Exploring the New Horizons; Scanning the New Development.

I. INTRODUCTION

Ever increasing diversity of digital technologies and use scenarios are continuously challenging digital preservation practices and constantly moving the goal post for the preservation action. In this paper we present a case study that required us to revisit the two fundamental notions in the digital preservation: the preservation of significant properties and the management of access and reuse.

Originating from a highly regulated sector that involves pharmaceutical, life sciences and bio-analysis organizations, the use case includes strict guidelines on the data retention and reproducibility of archived studies. Similar to other archiving practices, long term archiving of digital records is managed through a combination of format standardization and interoperability of both digital record formats and content management systems. However, the raw data that arise from research experiments have to be stored in the original format supplied by specific instruments (Figure 1).

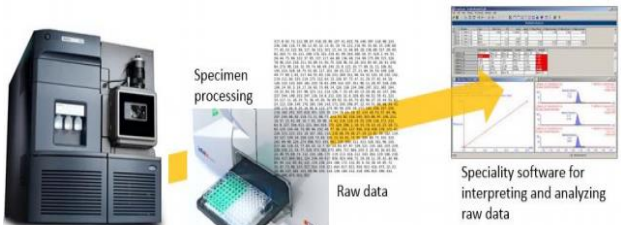

Figure 1 Raw data is produced by specimen processing and processed using software designed to support specific data analyses. The instrument and software installation are subject to an extensive calibration and validation process

The collection and handling of research data during the operational phase are subject to strict data integrity regulations that, in the archiving phase, translate into well-defined procedures for data deposit, meta data management and regular file fixity checks. Raw data must stay immutable (Figure 2). The unresolved issue, however,

17th International Conference on Digital Preservation

iPRES 2021, Beijing,China.

DOI: 10.1145/nnnnnnn.nnnnnnn

is the reproducibility and validation of the reported study results.

Reliable reconstructions of studies depend on the integrity of the software installations used to perform data analyses. Thus, both the data integrity and the software integrity requirements affect the preservation practices as they must enable the organization to meet evolving regulations and support regular compliance audits (normally every couple of years). However, there is another layer of complexity. While the study records and raw data are stored in the archive, the operation of the software lies outside the area of an archivist's competence. Indeed, the studies are reconstructed by scientists. Similarly, the management of the software installations, particularly software reliant on legacy operating systems, lies outside the area of an archivist's or a scientist's competence and must be addressed by IT specialists in a principled and well documented manner.

OECD SERIES ON PRINCIPLES OF GOOD LABORATORY PRACTICE AND COMPLIANCE MONITORING
Number 17

Appendix 2: Glossary

Data (derived data) Derived data depend on raw data and can be reconstructed from raw data (e.g., final concentrations as calculated by a spreadsheet relying on raw data, result tables as summarized by a LIMS, etc.).

Data (raw data) Data (raw data) may be defined as measurable or descriptive attribute of a physical entity, process or event. The GLP Principles define raw data as all laboratory records and documentation, including data directly entered into a computer through an automatic instrument interface, which are the results of primary observations and activities in a study and which are necessary for the reconstruction and evaluation of the report of that study.

Figure 2 Definitions of derived and raw data specified in the glossary of the OECD guidelines [13] (p31).

This separation of concerns and roles led to the concept of *Executable Archive* that extends the notion of a traditional archive with a *Software Library platform and services* that (1) host the collection of validated software installations, (2) provide secure connections to data repositories, and (3) enable access to software and data in accordance with the regulatory requirements. We illustrate the key aspects of the Executable Archive model by describing the process of *software transition* from operational use to a '*data reader*' use. The software transition puts an emphasis on both (a) the process of software installation and validation, i.e., the reader set-up and (b) the expert inspection of the data processing outcomes. Thus, the specification of the significant properties is split across the software preparation process and the data analyses characteristics.

While the Executable Archive approach is motivated by practices within a specific sector, the need for regulatory compliance and research reproducibility are broadly recognized. Data retention and reproducibility requirements are present across industry sectors, from fintech to aerospace [16,17]. While the General Data Protection Regulation (GDPR)expects organisations to create data retention policy, it does not specify the retention periods and those will vary across industries and type of data (e.g., 3-10 years in financial sectors [16] to 50 years for the design data in the aerospace industry [17]. Here we use a generic attribute 'long-term' to mean the longest retention period required in any specific sector. At the same time, government funding agencies are promoting open research data repositories and research hubs to enable reusability of data and maximize the impact of research investment [18,19]. Such initiatives typically provide tools for ingest, documentation and search of research data but still lack clear guidelines and requirements on validation and reproducibility of results.

II. Background

A. *Data Collection and Technology Management*

The process of data gathering and analysis starts with instruments and specimen processing (Figure 1). Interaction with the raw data is facilitated by specialized software, a key enabler of the data interpretation and analysis. Reports from the experiments are stored as evidence of observations, findings, and conclusions. Any changes to the software or the environment within which the software operates may affect the results. For that reason, the technology vendors are concerned with both (1) the implementation of the software and (2) the environment in which the software runs. It is common for vendors to supply a dedicated PC with pre-installed software to be used for processing data in the lab. They provide extensive service support and software upgrades that must be tested when deployed. The problems arise when the instrument and the software are no more in operational use either because the technology is discontinued or because the organization has changed the technology provider. In both cases, the instruments and the software are decommissioned. That leaves the archived data without a supported software.

B. *Regulations*

The importance of raw data and validation of research outcomes is emphasized by the *Good Laboratory Practices* (GLP) that the organizations must adhere to. The *Organisation for Economic Co-operation and Development*

(OECD) works closely with the professional community on the guidelines for complying with GLP regulations. Two aspects are particularly key to our discussion: the requirement for reproducibility of research directly from raw data (Figure 3) and a recognition that the software is important for the readability and validation of archived data and therefore must be managed as part of the archiving practices (Figure 4).

OECD SERIES ON PRINCIPLES OF GOOD LABORATORY PRACTICE AND COMPLIANCE MONITORING
Number 15

Advisory Document of the Working Group on Good Laboratory Practice

Establishment and Control of Archives that Operate in Compliance with the Principles of GLP

The archiving of records and materials generated during the course of a non-clinical health or environmental safety study is an important aspect of compliance with the Principles of Good Laboratory Practice (GLP). The maintenance of the raw data associated with a specific study and the specimens generated from that study are the only means that can be used to reconstruct the study, enabling the information produced in the final report to be verified and the compliance with GLP of a specific study to be confirmed.

Figure 3 Excerpt from the OECD guidelines for establishment and control of archives and raw data storage for compliance with Good Laboratory Practices (GLP) [13] (p9).

OECD SERIES ON PRINCIPLES OF GOOD LABORATORY PRACTICE AND COMPLIANCE MONITORING
Number 17

Advisory Document of the Working Group on Good Laboratory Practice

Application of GLP Principles to Computerised Systems

3.2 Data and storage of data

75. Hardware and software system changes must allow continued access to, and retention of, the data without any risk to data integrity. When a system or software is updated, it must be possible to read data stored by the previous version or other methods must be available to read the old data. Supporting information (e.g. maintenance logs, calibration records, configuration etc.) which is necessary to verify the validity of raw data or to reconstruct a whole study or parts of it should be backed-up and retained in the archives. Software should be retained in the archive if necessary to read or reconstruct data.

Figure 4 Excerpt from the OECD guidelines for application of GLP principles to computerized systems [1] (p20).

C. *Data Integrity and Software Integrity*

In order to support organizations in meeting regulatory requirements, we had to consider operational practices that led to the production of data and archived studies. These practices are shaped by concerted efforts to maintain the data integrity throughout all the aspects of the research work. For data produced using computerized system that inevitably means rigorous management of hardware and software to ensure the quality of collected data. It is therefore helpful to consider data integrity and software integrity together (Figure 5).

Data Integrity is of ongoing concern and a matter of constant improvement, from increased security and interoperability to a reliable management of data provenance and digital signatures. The community is actively pursuing interoperable XML-based formats for

Data Integrity

Data Integrity is the extent to which data are complete, consistent and accurate throughout the data lifecycle.

- **Complete** – All the required data needs to be collected and stored
- **Secure** – Data must not be destroyed
- **Unaltered** – Data must not be changed
- **Confidential** – Data must not be disclosed to unauthorised individuals
- **Usable** – Study metadata needs to provide context for experts reproducing study results.

Software Integrity

Software Integrity is the extent to which a software installation is functional, reliable, and usable throughout its lifecycle from operational use (i.e., applied in new studies) to study reproducibility (i.e., applied to archived data).

- **Functional** – Software installation must stay operational
- **Unaltered** – Software must not be changed
- **Secure** – Software installation must not present risks from cyberattack and confidentiality breaches
- **Accessible** – Software installation needs to be easily accessed for repeated use.

Figure 5 Data Integrity and Software Integrity definitions.

representing raw data and data analysis in order to automate encryption/decryption of data files as the data is moved between different applications for various types of analyses. That work is ongoing [14]. Once a study is completed, the researchers transfer data for archiving and preservation to the Central Archive. The data is regularly checked for bit-rotting issues by conducting check-sum validation of data samples on a monthly basis.

Software Integrity, on the other hand, has not been of much concern since operations are supported by a careful and comprehensive validation of instruments and software at the time of the technology deployment and upgrades. That ensures that the software stays performant, secure and consistent. However, when the software is decommissioned the software care stops and that led to a number of *ad hoc* approaches to ensure a sustained use of software, from creating an image of the full computing environment to re-installing the required software within a suitable computing environment. No principled ways of managing the software in the archiving phase has been established.

D. Summary

The bio-analysis research use case highlights two key issues:

1) The success of the preservation process is dependent on the data file fixity but the preservation and demonstration of the significant properties are subject to the software integrity, i.e., ability to re-compute the data and reliably reproduce the results.

2) The regulatory requirements mandate the archiving of original software alongside the data, clearly recognizing that the capability of data presentation and data analyses is not in the file format but in the computation of the raw data files.

One may argue that the preservation of the final study reports, e.g., using a standardized rich file format with imbedded data, should be an alternative approach, assuming that there exist reliable and regulated standardized readers. Unfortunately, normalization of raw data and data analysis formats across instrument analyses is difficult to achieve, if not infeasible. Furthermore, one cannot underestimate the challenge of proving that a substitute software (reader) can reliably produce the same results as the original, nor can we easily determine the impact that invalid results may have. The latter was recently illustrated in a highly reported case of Public Health England, missing to account for thousands of Covid cases due to a software versioning problem [2].

In the following sections we first reflect on the related work in digital preservation and management of software and then describe the Executable Archive approach to the long-term maintenance and validation of Analyst 1.4.2 (Sciex) installations required for accessing and validating pre-clinical study data.

III. Related Work

Importance of digital objects authenticity and preservation of software has been recognized by the digital preservation community and led to research efforts dedicated to developing effective methods. Here we provide a brief overview of the past work relevant for framing our research effort and contributions.

A. Preservation of Significant Properties

The term 'significant property' has different interpretation in literature. Open Archival Information System (OAIS) standard [4][15] defines it as an information property that is necessary for preserving the information content across any non-reversible transformation, while PREMIS [5] refers to it as a specific set of meta-data attributes required for rendering a file or a digital object. Both definitions emphasize the link between significant properties and authenticity of digital artifacts, but also the subjectivity of their choices.

The subjectivity is a result of a specific domain's assumptions of what is necessary or worth preserving. For example, preserving colors may deem important for an art eBook but not necessary for a history eBook in which case it is sufficient to preserve words, punctuation and paragraph separation. Moreover, in Digital Arts, the definition of significant properties is expanded outside of the file-related attributes to include behaviors, rules of engagement, and visitor experience amongst others [3].

In the context of our use-case, the preservation of significant properties relates to the ability to reproduce a scientific study rather than a digital object. The data analysis is instantiated by re-computing the raw data. One may thus argue that, according to the OAIS interpretation, the only significant properties are the stored results of the study or their selected subset; more precisely, the input- output dataset of the archived study. However, this interpretation does not take into account the requirement of preserving the operational environment. In that context the PREMIS meta-data interpretation of the significant properties is more suited, with relevant attributes spanning the characteristics of data, network and software components of the preservation environment.

As suggested by Matthews et al [6}, besides the significant properties of the input dataset, e.g., attribute-value pairs and instance numbers, one needs to consider additional data such as characteristics of the network (e.g., the security protocol) and the software (e.g., functionality, composition, ownership and other properties defined in [6]). In our use case, the necessary meta-data about the software are included and verified through specific 'qualification' procedures (Figure 8), before the software is transitioned to the Software Library platform. The qualification procedures are closely linked with the practices of maintaining software during its operational use when it was critical to ensure that the manufacturing process produced quality data. The goal of the qualification procedures is to guide the installation process so that the archived software installations produce outputs consistent with a predetermined quality.

The choice of significant properties remains a major research question for the preservation community in various domains, including digital games [11], and is a

pre-condition for selecting an optimal preservation strategy.

B. Validation of Software Installations

The efforts required to enable stable installations and provide ongoing maintenance, to keep the software operational, results in a significant cost. While in other industries the maintenance cost is estimated to be between 10 and 25 percent of total operating costs [7], software maintenance contributes to a much higher percentage of the total software life cycle cost (e.g., 66% quoted in [8]).

In fact, the high cost of maintenance has been identified as one of the key external factors that contribute to the software aging [9]. According to the same study the software aging metrics include not only performance, usefulness, business demand, environment and technology change but also a need to retain and train experts. The same applies beyond the typical software use period, i.e., when both the data and the software need to be archived. This need is heightened with premature software aging as software release cycles are becoming shorter and shorter [10].

Development of service-based software models, replacing the product view of the software, has been recommended in late nineties [11] as a step forward in reducing the cost of ownership. Since then various 'as a Service' models have emerged such as SaaS, PaaS, IaaS to mention a few. The Executable Archive framework is, in effect, a software-as-a-service model, with fully managed hosting of virtualized software that belongs to the user, i.e., the user's organization.

C. Long Term Software Management

Aging of software typically involves two technical factors, the deteriorating hardware and unsupported, i.e., insecure operating system. Virtualization can assist with both. The technique allows a user to execute their software application in a different operating environment from the host system, thus taking advantage of the host hardware. This has a broader applicability, addressing the issues of incompatibilities of software programs with different operating systems. For example, software such as Microsoft Project that does not have MAC OS binaries can be run on top of a VMWare virtual machine on a MAC machine. By reducing hardware/software dependencies, virtualization enables cloud-based provision of services and more efficient and productive software maintenance [20]. In other scenarios it assists with prolonging the life of installations that involve software, such as modern sculptures and digital arts, where software is an integral part of the artefacts [3].

The term virtualization is sometimes used interchangeably with emulation. There are similarities between the two methods as they both allow the code originally developed for one system to execute on another. However, they differ in several key technical points:

- Emulators interpret the source code into the CPU instructions of the host machine, while in virtualization, the original code (binaries) is executed in a 'container' process that provides a bridge between two operating systems.
- Emulators are slower compared to virtualized applications.

From our perspective, the most important difference is that virtualization aims to provide a generic execution environment for any application (e.g., enables any application that requires Windows environment to run on a MAC server). Emulation, on the other hand, provides a bridge between a specific application and the host hardware, e.g., enables an old Atari game to run on a Windows laptop.

However, the virtualization software itself is subject to aging, i.e., lack of support. In our use-case we adopted Xen virtualization provided by Citrix which has an open-source counterpart. That helps mediate some of the risks of virtualization. Generally, the risks of virtualization need to be carefully considered [12] in order to take measures to mitigate them. For example,

- Licensing and cost issues, as the license is required for all virtualized operating systems, and a suitable Range of Host Platforms and Operating Systems might need to be supported
- Performance might be an issue in the environments where near real-time performance is expected.
- Aging and maintenance of the virtual platform itself need to be carefully monitored and planned for.

IV. Case Study: Reproduction of Archived Research Studies in Bio-Analysis

In this section we describe the practices developed to ensure reconstruction of archived research studies by a bio-analysis researcher in order to meet the GLP compliance audits [1]. We focus on supporting the act of reproducing a specific result. However, it is worth mentioning that the archiving of study data follows a

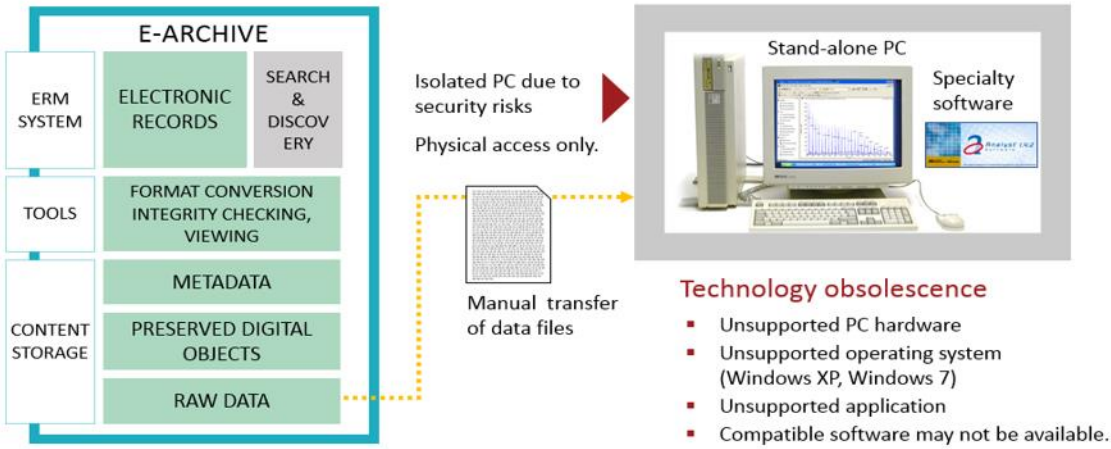

Figure 6 Components and data access in traditional 'PC with software installation' preservation case

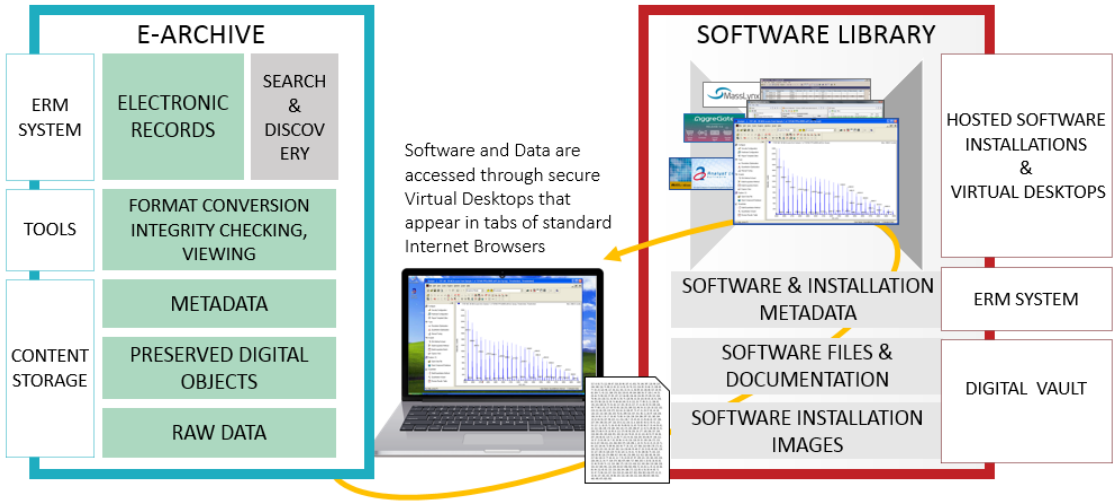

Figure 7 Components and data access in the proposed preservation framework

well specified procedure and a shared practice adopted by researchers and archivists, who are involved in the study deposit process. The deposited data involves metadata that enables researchers and archivists to locate the specific study very efficiently within the record management system. The system includes contextual information of the study and the accompanying documents in a standardized format, most often PDF. The reported graphs and statistics, derived from the raw data analysis need to be reproduced. The stored representation, e.g., a report in the PDF file format, is a different digital object from the raw data files. The raw data file characteristics will be revealed only through the computation and rendering of the results on the screen. Thus, the emphasis is on the properties of the software and therefore on the well-controlled process of software installation and validation. This required special care as the supporting operating system is Windows XP SP2, thus no longer supported and insecure.

The second aspect is the separation of the virtualized software, hosted on the Software Library platform, from the archive repository. Since legacy software installations cannot be exposed, i.e., connected to the organizational network, one has to either isolate both, the archive and the software installation, or extract data from the software repository and bring it into the VM environment. The latter approach was deemed more appropriate. Thus, a support for the data export and transfer had to be carefully designed and implemented.

Both of these present novel contributions to the preservation practices in general and improve preservation of scientific results in particular. Implementation follows a software-as-a-service model with fully managed and remotely used collection of virtualized software installations. Access to the archived data repository is configured for secure transfer and use within the running software sessions. The concept is applicable to general archives with data integrity and access requirements.

A. *Archived data readability*

Once a research study is completed, raw data, analysis data and documentation are placed in e-Archive. Archived data accumulates over time. On the other hand, as new instruments are adopted, the previous ones are decommissioned. The software would normally be decommissioned at the same time but is needed to read the archived data, for decades. In our instance the software package Analyst 1.4.2 produced by Sciex had to be decommissioned as the organization stopped using the corresponding instrument. The studies were produced in the period from 2006 and 2015 when a different product was adopted. Thus, readability of all the studies over the period of 9 years is affected if the software is not in use anymore.

Up to that point, the data readability was achieved by maintaining an isolated PC with the original copy of the Analyst 1.4.2 installation. This is a common practice but not sustainable solution due to possible hardware failure. Thus, one needs be prepared to re-install the software on a compatible machine. At that point one may as well eliminate the dependence on the hardware component and adopt virtualization.

The isolation of the PC due to the insecure operating system affects the way the compliance audit can be conducted. Namely, if the archive is on the network for ease of use and management, then the PC should not directly interact with it. Therefore, the archivist needs first to export data and place it on a medium that can be read by the PC, e.g., a USB stick or external hard drives. This transfer of data will always be an issue. Two particular aspects are of concern: (a) one has to guarantee that the data is not changed during transfer and (b) data should not be left on the portable devices or on the PCs due to data protection and privacy regulations.

The archived data readability problem can then be defined as two tasks (a) create an installation that is for all practical purposes an equivalent to the PC installation and (b) provide a mechanism for easy input of data into the virtual machines that uses the legacy operating system (Figure 7).

V. TECHNICAL SOLUTION AND PRACTICES

A. *Software Installation and Validation Approach*

In a private data centre, we

- Create a sandboxed VM environment to enable installations of Analyst 1.4.2 software with WinXP SP3 operating system.
- Enable upload of the software into the Software Library environment
- Follow the original installation instructions, applied to the installation of the software on the lab PC. These instructions are referred to as Installation Qualification (IQ).
- Document the process of installing the software in the VM. The new documentation is referred to as Software Library IQ (SL-IQ) indicating that the installation is virtualized.

This first part of the installation process represents a critical task of addressing and documenting all the adjustments of the archived installation in comparison with the original installation, e.g., single-user vs multi-user installation, security settings for a stand-alone vs networked installation, user authentication, software activation, and related. If the rest of the process proves to be successful, SL-IQ becomes a blue-print for all other subsequent installations that may have to be done in the future.

Analyst 1.4.2 software by ScieX has been used by a pharmaceutical organization since 2006.

The instrument set-up and the software installation followed the best practices and produced documentation

- **Installation qualification** (IQ)
- **Operational qualification** (OQ)
- **Performance qualification** (PQ)
- **Re-qualification** after the initial IQ, OQ, PQ and in accordance with a user's Standard Operating Procedure (SOP) requirements.

Figure 8 Virtualized legacy software Analyst 1.4.2 has been originally installed in 2006 and virtualized in 2019 using the same software qualifying procedure.

The next stage requires researchers to test the features of the installed software in the VM. That involves specifying the task and setting up the appropriate Virtual Desktop configuration to support the task. The involves a researcher's effort to (a) review the documentation of the original software validation, referred to as Operational Qualification (OQ) documents and (b) select the set of software features that support the study reconstruction task and must be tested. The result of this process is SL-OQ, i.e., operational qualification criteria for the evaluation of the virtualized installation of the software.

The researchers

- Describe the study reconstruction steps by selecting a sample data set.
- Perform the study reconstruction steps and compare with the OQ documentation and expected outcomes.

In addition to the complete task qualification process the researchers also create a short test that can be used just to test that the software has not changed between usage. Similar tests are performed on the original software installation from time to time and is referred to as Performance Qualification (PQ). Thus,

- Researchers decide on the minimal set of interactions with the virtualized software that should be used to establish that the Software Integrity is intact
- The resulting set of actions is referred to as Software Library PQ and will be applied every time the software is used and before importing the real data.
- Document the outcomes of the SL-PQ based on the software screenshots. This document will be used as a reference in all the use scenarios, including the compliance audits.

B. Software installation and testing of Analyst 1.4.2

For Analyst 1.4.2 we followed the described approach and successfully created SL-IQ, SL-OQ and SL-PQ procedures. Figure 9 describes the three stages.

1. DEV stage involves the Sandboxed VM, using SL-IQ instructions for Analyst 1.4.2, and ensuring that the installation is as close to the original as possible. Controlling the installation process serves as assurance that even the features that have not been tested explicitly are likely to stay functional as with the original installation.
2. TEST stage involves Virtual Desktop access to the Analyst 1.4.2 that enables the user to use the data attached to the VM to apply SL-OQ and SL-PQ procedures. All the outcomes are compared with the same test run on the PC in the Lab which is still functional.
3. PROD stage involves the final release of the software for use on the Software Library platform. The testing of the PROD environment is conducted by the IT staff to confirm the performance parameters that were already established in the TEST phase which relate to the speed of upload, movement of data on to the Analyst 1.4.2 VM, decompression and checksum testing of the data.

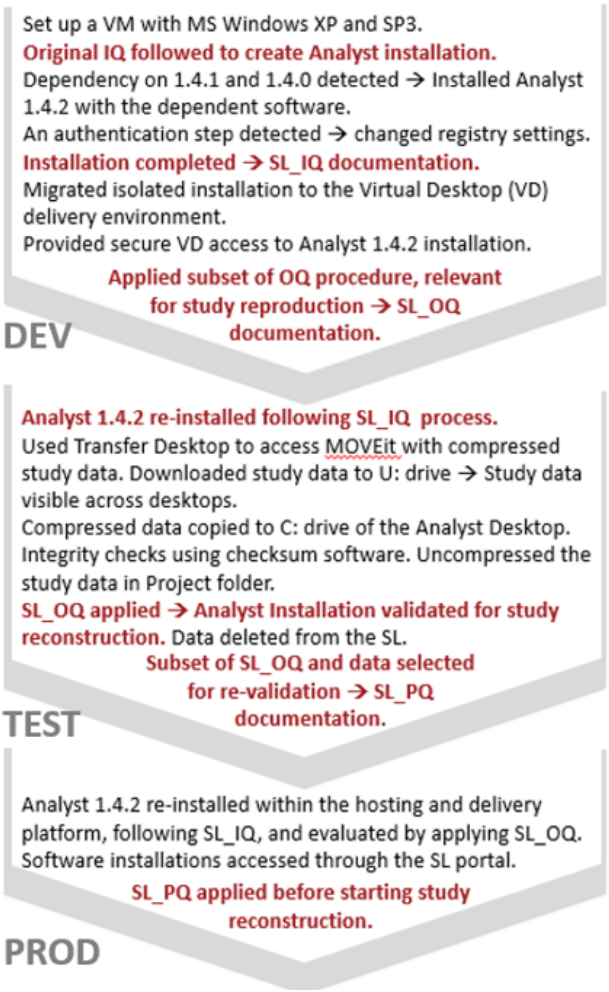

Figure 9 Phases in the installation and validation process.

C. Study Reconstruction Test

Full study reconstruction test of Analyst 1.4.2 (Figure 10) was performed using data exported from the archive. It contained a large collection of studies which could not be separated into individual study file due to the organization of the files by the Analyst Software.

Particularly important was to ensure that all the audit files associated with the data can also be viewed in the software installation. The researchers advised that the audit files can be viewed only if the data were placed on the specific path, i.e., stored on the C: drive. Thus, the IT staff had to consider the speed of data management: upload of the data into the Software Library platform, checksum verification of the zipped file, moving the data to the destination i.e., C: drive and then decompressing the data.

The data size of the Analyst 1.4.2 archive was a 1GB: 9GB uncompressed. compressed file. Testing of the installations involved the SL_PQ procedure, performed using a copy of archived data: 1Gb compressed; 9Gb uncompressed. The transfer from using MOVEit data

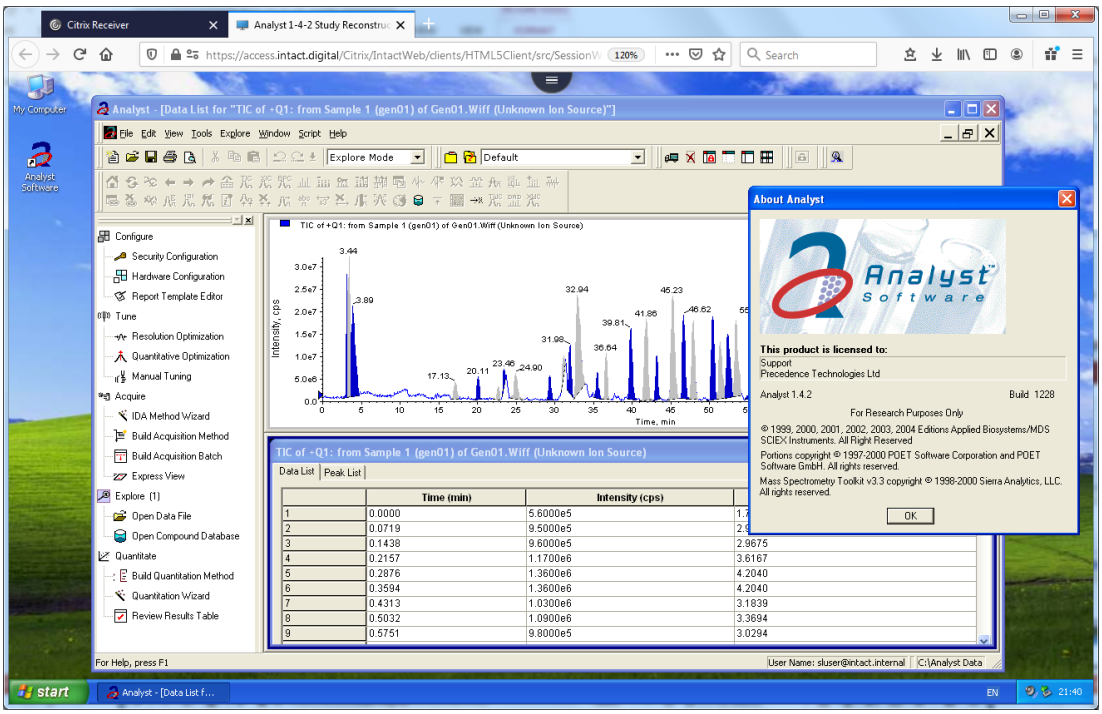

Figure 10 Reconstruction of research study is enabled using a validated virtualized installation of Analyst 1.4.2. The figure shows a file from the Analyst 1.4.2 data sample that comes with the software release. The file is dated Dec 2001. The Analyst 1.4.2 version is from 2004.

deposit was 1 min. Within Software Library network copy, to C: drive of the Analyst Desktop took about 2 min. The checksum was < 20xec and the decompression about 10min. Thus, within less than 14 min, the large data collection was ready for inspection. Changing the order of data management, e.g., uncompressing the file before moving to the C: drive increased the time by ~ 40min.

VI. Conclusions and Future Work

In this paper we presented a case of archiving practice that requires a different approach to defining and enforcing the preservation of significant properties. Since the research study must be reconstructed from raw data, the reproduction of results requires re-computation of the data. Thus, it is the software properties that determine the outcome. That, in turn, calls for introducing Executable Archives as an extension of the traditional archive with a Software Library platform that hosts virtualized installations of the required 'reader software'.

The validation of the virtualized software installations closely follows the software installation practices that are enforced by the companies deploying and maintaining the software during its operational time span. These procedures are adapted to the VM hosting environment and serves as the mechanism for maintaining the software integrity of legacy installations over time.

We demonstrated the technical feasibility of hosting and remote use of installations even when relatively large files need to be moved into the environment. The method is effective, fully compliant with organizational policies and aligned with established validation practices. It does not require any changes to the data or software. In fact, it is devised to preserve both Data Integrity and Software Integrity.

Going forward, we advise to optimize the process further by adding software to the Software Library at the time it is first deployed and subsequently upgraded. That has two advantages: (1) the validation process need not be performed (again) at the time of software decommissioning and (2) the Executable Archives with up-to-date Software Library is always aligned with the archived data and content.

References

[1] OECD Series on Principles of Good Laboratory Practice and Compliance Monitoring No. 17: Application of Good Laboratory Practice Principles to Computerised Systems, OECD, Paris, 2016.

[2] BBC News, "Covid: Test error 'should never have happened' – Hancock" https://www.bbc.co.uk/news/uk-54422505

[3] A. Ashe, P. Falcao, and B. Jones. "Virtualisation as a Tool for the Conservation of Software-Based Artworks." In Proceedings of the 11th International Conference on Digital Preservation (IPRES), Melbourne, Australia, October 6-10, pp. 83-90. 2014.

[4] Consultative Committee for Space Data Systems. (2012). Reference model for an open archival information system (OAIS). RECOMMENDED PRACTICE CCSDS 650.0-M-. MAGENTA BOOK June 2012 CCSDS Secretariat.

[5] PREMIS Data Dictionary for Preservation Metadata, version 3.0 (June 2015), http://www.loc.gov/standards/premis/

[6] Matthews, B. mcIlwrath, B., Giaretta, D., Conway, E., 2008, The Significant Properties of Software: A Study . JISC

[7] Mckinsey& Co., Planning to fix: improving maintenance efficiency, September 1, 2012 https://www.mckinsey.com/business-functions/operations/our-insights/planning-to-fix-improving-maintenance-efficiency

[8] Yip, S. W., & Lam, T. (1994, December). A software maintenance survey. In Proceedings of 1st Asia-Pacific Software Engineering Conference (pp. 70-79). IEEE.

[9] Abdullah, Z. H., Yahaya, J. H., Mansor, Z., & Deraman, A. (2017). Software Ageing Prevention from Software Maintenance Perspective–A Review. Journal of Telecommunication, Electronic and Computer Engineering (JTEC), 9(3-4), 93-96.

[10] Yahaya, J. H., Abidin, Z. N. Z., & Deraman, A. (2015, July). Perspective and perception on software ageing: The empirical study. In 2015 10th International Conference on Computer Science & Education (ICCSE) (pp. 365-370). IEEE.

[11] Bennett, K. H., & Rajlich, V. T. (2000, May). Software maintenance and evolution: a roadmap. In Proceedings of the Conference on the Future of Software Engineering (pp. 73-87).

[12] McDonough, J.P., Olendorf, R., Kirschenbaum, M., Kraus, K., Reside, D., Donahue, R., Phelps, A., Egert, C., Lowood, H. and Rojo, S., 2010. Preserving virtual worlds final report.. Available at: http://www.ideals.illinois.edu/bitstream/handle/2142/17097/PVW.FinalReport.pdf

[13] OECD Series on Principles of Good Laboratory Practice and Compliance Monitoring No. 15: Establishment and Control of Archives that Operate in Compliance with the Principles of GLP, OECD, Paris, 2007.

[14] Celebi, I., Dragoset, R.A., Olsen, K.J., Schaefer, R. and Kramer, G.W., 2010. Improving interoperability by incorporating UnitsML into markup languages. Journal of research of the National Institute of Standards and Technology, 115(1), p.15.

[15] Giaretta, David, Brian Matthews, Juan Bicarregui, Simon Lambert, Mariella Guercio, Giovanni Michetti, and Donald Sawyer. "Significant properties, authenticity, provenance, representation information and OAIS information." (2009).

[16] FCA Handbook (2021) Available at: https://www.handbook.fca.org.uk/

[17] International Aerospace Quality Group standards https://iaqg.org/

[18] The Open Research Data Task Force (UK) https://www.universitiesuk.ac.uk/policy-and-analysis/research-policy/open-science/Pages/open-research-data-task-force.aspx

[19] Gates Open Research https://gatesopenresearch.org/

[20] IBM (2021) Virtualization https://www.ibm.com/cloud/blog/5-benefits-of-virtualization#:~:text=Five%20benefits%20of%20virtualization.%201%201.%20Slash%20your,to%20be%20more%20green-friendly%20%28organizational%20and%20environmental%29%20