Digital Preservation Challenges on Software Life Cycle

José Barateiro
LNEC
Lisbon, Portugal
jbarateiro@lnec.pt

Daniel Draws
SQS AG
Cologne, Germany
daniel.draws@sqs.com

Martin Alexander Neumann
KIT
Karlsruhe, Germany
mneumann@teco.edu

Stephan Strodl
SBA
Vienna, Austria
sstrodl@sba-research.org

Abstract—TIMBUS is a partially EU-funded research project in the area of digital preservation. The project aims at a holistic view on the preservation of business processes over long-term periods. This view includes the business software as a well as all related context information for re-use at a later point in time. Digital preservation ensures continued access to digital information over time (e.g., 5, 15, 25 or 50 years). The long-term planning horizon of preserving relevant parts of a business processes raises a number of new research challenges for software design, development and maintenance.

Current service-oriented systems are using distributed services from different providers located at different geographical locations. Workflows and business processes are volatile in terms of services disappearing and fundamental changes in technology. TIMBUS researches on methods, processes and tools to ensure continued access to information based on preserving relevant data and their context (e.g., services, software and the data's meaning). The project aims at identifying, capturing and preserving relevant information and their dependencies necessary to understand the preserved business processes in a remote future.

In this paper we describe the primary objectives of the TIMBUS project with special focus on challenges caused by the preservation of software.

Index Terms—Digital Preservation, Process Preservation, Business Processes, Software Quality

I. INTRODUCTION

The ephemeral character of digital data and information leads into serious issues in preserving information over the long term [1]. Digital preservation aims at ensuring the operability and usability of digital information. The long term preservation of business process raises a number of new challenges with respect to software development and maintenance. The TIMBUS[1] project aims at providing methodologies, tools and guidelines that support the preservation of existing business processes and the creation of new "preservation-aware" processes. Concerns addressed in TIMBUS range from legal and economic aspects to technical aspects of software development and the emulation (or virtualization) of complex software systems.

Business processes are an orchestration of different services. Many of these services are operated by different service providers. Processes are volatile in terms of services disappearing and fundamental changes in technology. Processes of vital interest in a remote future need to be preserved.

Their functionality, usability, integrity and authenticity need to be guaranteed. Examples for business processes are present in domains such as civil engineering, e-Science, multimedia productions and particle physics. Changes of technology (e.g., new formats or standards), changes of the environment (e.g., new legal obligations) and the disappearance of services need to be addressed for ensuring the usability of the process and its software in the long run.

Current maintenance approaches for software are dealing with online (and active) systems. Changes in the environment, technology, services and data potentially have immediate effects on the system functionality. For preserved software stored off-line in an archive these changes can remain concealed for years. New approaches for active management and maintenance of preserved information systems are required. The risk of irreconcilable inconsistency of the software system caused by technological obsolescence or unavailability of essential services and software components need to be addressed.

Preservation of software systems raise new challenges for all stages of the software life cycle. Starting from the design phase, well-documented and clearly designed structures of software reduce the dependencies on actual implementation technology. Other examples are requirements on the software quality raised by the fact that source code should be understood in the future. Maintenance over the long term can include the replacement of obsolete technologies, such as runtime environments, operating systems, networking infrastructures, etc. Other research issues that are addressed within the TIMBUS project include, amongst others, authenticity of the preserved information, legal/contractural/regulatory issues and risk management with respect to business process preservation.

A. Software Quality

Significant progress has been made over the last years in the understanding of software quality. ISO norms like ISO 25010 ("Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models" [2]) and the well established ISO 9126 [3] provide a suitable guidance for continuous maintenance of a software product [4].

With the need of preserving digitally born information new requirements came up for software and software quality. In order to transform data into information some kind of

---

[1]http://www.timbusproject.net

representation information is needed. A powerful representation information is the software that generated the data. For preservation of information for decades it is obvious that software could need to be maintained for a long time. The difference to the established maintenance approaches is the fact that archived software stays unchanged for a long period. The development process has often to be re-instantiated in a completely different and unknown environment.

Current software quality approaches focus on the quality of source artifacts. For the long term preservation of software system with a holistic view all artifacts used in a development and deployment process are needed [5]. Besides the technical preservation of the software system, the long term focus of business processes in question needs to be aware of different changes in the environment (such as change of legal aspects over time or transfer of tacit knowledge). These changes can raise new requirements for software quality.

B. Digital Preservation

Digital Preservation—ensuring continued access to digital assets—is traditionally focused on data-centric information. A number of research initiatives have addressed preservation of digital objects, such as office data, video or more complex data structures such as databases. An overview about relevant research projects within the ICT program of the European Commission is given in [6].

The data centric view leaves out important aspects of execution aspects such as processing, analyzing, transforming and rendering. Digital preservation of processes and services is a novel approach. Workflows and business processes often rely on distributed and service oriented architecture. Unavailability of data or services can lead to loss of information resulting in negative effects for the businesses they enable.

The different technologies of system components, their dependencies and ongoing developments raise a number of new challenges for digital preservation. TIMBUS investigates to identify and resolve dependencies under given constraints to ensure re-executability of processes in remote future.

II. PROJECT OVERVIEW

A primary motivation for TIMBUS is the declining popularity of centralized in-house business processes maintained and owned by single entities. The presence of Software as a Service (SaaS) [7] and Internet of Services (IoS) [8] means business processes are increasingly supported by service-oriented systems where numerous services provided by different providers located in different geographical locations are composed to form service systems which will continue evolving. Besides the advantages of SaaS and IoS, there is the risk of services and service providers disappearing (for various reasons) leaving partially complete business processes.

Successful digital preservation of business processes requires capturing sufficient detail of a business process and its context to be able to re-enable its original behavior at a future date, involving potentially different participating parties, different enabling technologies, different system components

(hardware and software), changed services by different service providers or differences in other aspects of the context of the business process. Digital preservation of business processes and services therefore requires that the set of activities, processes and tools that ensure continued access to services and software necessary to produce the context within which information can be accessed, properly rendered, validated and transformed into knowledge can be preserved.

Traditional digital preservation approaches have a focus on preserving digital objects and their context [9]. However, from a TIMBUS perspective, traditional methods of digital preservation do have a too restricted perspective on context that they take into account. Assuming that a traditional method of digital preservation provides the ability to preserve system components and entire systems in addition to digital artifacts, often, it would not be clear at a given time, which information would be required later to restore the system. What is a clear boundary of technical feasibility today can be a parameter not accounted for in the future.

A brief example is computing speed. Consider a digital preservation time t_0, where we have a network connection whose speed is clearly below the speed of a database lookup. Furthermore, we assume there is a race condition: our service internally sends a request to a remote server and concurrently, at the same time looks up a required parameter in the database. As the database access is always faster than the network request, we obtain the parameter before the answer from the remote server arrives. If we exhume the digitally preserved system at a time t_1 in a virtual environment, we might get a system in which the simulated network is faster than the database access and the results might not be the same that the original system would have produced. Capturing such constraints is therefore crucial to prevent failure in exhumation of a business process.
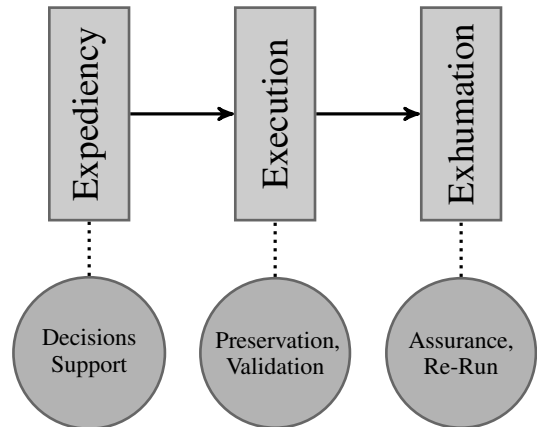

Fig. 1. Process for Digital Preservation of Business Processes in TIMBUS

A. Goals

The preservation process proposed in the TIMBUS project consists of three stages of digital preservation effort as shown in figure 1. To support the digital preservation of business

processes, TIMBUS has a number of objectives which are illustrated in the following in the context of TIMBUS' three stages of digital preservation effort:

Expediency of digital preservation effort – establishing the risk of not preserving and the feasibility of digitally preserving business processes. Analyzing risk is a complex process requiring many sources of information to be collated and reasoned over. TIMBUS will develop methods and tools that provide a smart approach that will support autonomous or semi-autonomous decisions relating to (1) when to preserve (for example by automated deduction as envisioned in [10]); (2) what to preserve; and (3) how to maintain and test what has been preserved.

Execution of digital preservation process – performing the digital preservation of business processes. After the expediency has been established it is necessary to actually execute the digital preservation process. Today's services are deployed on multi-tier service platforms that are not yet engineered specifically with digital preservation in mind. TIMBUS will address two issues: how to re-engineer existing services for digital preservation and how to engineer new services for digital preservation. The project will also develop validation methods which will assert whether business processes are preservable now and in (simulated) future.

Exhumation of digitally preserved assets – re-running a digitally preserved business process. It must be possible to exhume and rerun a preserved business process. Furthermore, TIMBUS must provide continuous assurance that a business process (which has been digitally preserved) can be exhumed and re-run in (simulated) future.

III. DIGITAL PRESERVATION CHALLENGES FOR SOFTWARE AND BUSINESS PROCESSES

Research efforts in digital preservation have been mainly adopted in cultural heritage scenarios (e.g., libraries, archives, museums) where the digital objects to be preserved are typically composed of single artifacts such as media files and text documents.

However, information systems and software components are also threatened by several issues, including digital preservation threats [11] that can result in software malfunctions or unavailability [12]. Usually, these digital preservation issues are only addressed by maintenance activities, resulting in very expensive adaptations or replacement efforts. In fact, to change the paradigm from the reaction to problems to the prediction and planning, it is needed to incorporate digital preservation concerns from the very beginning of the software life cycle.

Approaches like Model Driven Architectures (MDA)[2] intend to provide the separation of concerns between the logical level of systems from its implementation, reducing the dependency on the technological infrastructure [13], [14]. Although this type of approach provides a method to simplify the deployment of systems in distinct technologies, they do not address the fundamental starting of systems' design, including

[2]http://www.omg.org/mda/

digital preservation concerns and requirements to address, among others, technological disruptions and changes in the surrounding environment.

In scenarios like business processes and software preservation, the concept of a single artifact, which is common in cultural heritage, makes little or no sense. In fact, in order to understand the required properties and execution context required to maintain the value of a business process or software component, it is critical to define the requirements that characterize each specific scenario. For instance, it can be acceptable to "just" preserve the software behavior (the original inputs produce the same outputs in the preserved solution), without any constraints on timing/speed (critical in games or simulated environments). Some cases can require the maintenance of look-and-feel and user interaction properties. Another example is the case where it is intended to re-execute a full process or orchestration of processes (e.g., for future evidence, to prove past research) that can depend on multiple services, data and configuration settings.

Depending on the specific scenario and future usage requirements, the digital preservation of business processes and software include specific requirements that raise several new challenges. These challenges increase when the objective is the digital preservation of business processes that can have dependencies on multiple levels, from high-level business dependencies to technological dependencies between software, hardware and data representations.

As happens in common digital preservation scenarios, it is required that future consumers are able to re-execute the software or process as its creators intended to, dealing with obsolescence threats. To provide the future re-execution, digital preservation solutions must assure the execution in multiple and heterogeneous systems (including unpredictable future systems). This requires preservation actions to maintain the physical representation (bit stream preservation), but also the logical and semantic levels [15]. Assuring bit stream preservation is relatively straightforward, where replication strategies can be used to maintain the bit stream, while logical and semantic preservation require continuous actions to assure that digital contents are still understandable.

Most research on actual solutions for logical preservation is focused on two prevailing preservation strategies: migration and emulation [16]. Migration refers to converting a digital object from one hardware or software generation to another generation or representation. For software, an example of migration can be the recompilation of a software component with a new compiler version. In contrast, emulation operates on the environment, recreating the functionality of systems (software and/or hardware) which is needed to overcome technological obsolescence.

The application of migration or emulation techniques to software components, produce modifications on the software component or on the underlying environment. Due to dependencies between software components and/or the related environment, the preservation strategy must take into account the effects produced by each technique, reducing the risk of

failing in digital preservation requirements.

However, applying migration and/or emulation to software and processes requires a coherent set of principles, methods and models that are used in the design and realization of business processes, information systems and their infrastructure. Indeed, the separation of concerns between those levels is desirable to control the complexity, reducing the risks entailed to the architecture design and evolution, coping with technological changes [17].

We consider the main challenges to preserve software used within the execution of business processes as:

- Recognize the need for digital preservation. Digital preservation concerns should be addressed from the very beginning of the software life cycle. Solving digital preservation related issues a posteriori often results in undesirable costs.
- Capturing the business process and/or software context. The context strongly depends on specific scenario requirements, raising several challenges on context capturing and modeling, but also on defining the relevant context information that must be kept to fulfill scenario-specific requirements.
- The nature of complex objects composed by software, its configuration settings and their overall environment. The task of understanding what are the boundaries of the required context information is an error-prone and costly task. Loose of context can drastically reduce the understandability and future use of any software component.
- Dealing with migration/mapping techniques for complex objects, including the software itself, may involve recompilation, library update, etc. The level of dependencies to distinct software, library and hardware components is critical to fulfill digital preservation.
- Define policies for digital preserving software components is a complex task due to the dependencies on several levels and heterogeneity of object types and their underlying representation.

IV. Conclusion and outlook

The TIMBUS research project fosters business continuity by digital preservation. It aims at providing methodologies, tools and guidelines that support the preservation of existing business processes and the creation of new "preservation-aware" processes. Expected outcomes of the project are amongst others:

- A risk model for business processes with respect to digital preservation.
- Monitoring facilities for risk factors with interfaces to enterprise risk data sources.
- A context model for business processes. The model documents the contextual information and its dependencies. Software tools will support the creation of instances of the model.
- Reasoning systems to detect changes in running business processes and to support decision making concerning their preservation.

- Preservation processes required to preserve business processes in the long term.
- Guidelines for "digital preservation-aware" process design and implementation (including software development).
- Long term storage facilities for business processes.
- Validation methods for integrity of preserved business processes.
- Overview of legal aspects for preservation of software services and their data.

The results will be applied to three industrial use cases within the project: software, service and systems engineering, civil engineering and large-scale scientific simulation and data analysis.

References

[1] T. Kuny, "A Digital Dark Ages? Challenges in the Preservation of Electronic Information," Sep. 1997.

[2] *ISO/IEC 25010 Systems and software engineering - System and software product quality requirements and evaluation (SQuaRE) - System and software and software quality models.*

[3] *ISO/IEC 9126-1:2001 - Software engineering. Product quality.*

[4] F. Simon, O. Seng, and T. Mohaupt, *Code-quality-Management. technische Qualitt industrieller Softwaresysteme transparent und vergleichbar gemacht*, 1st ed. Heidelberg: dpunkt-Verl., 2006.

[5] D. Draws, S. Euteneuer, D. Simon, and F. Simon, "Short Term Preservation for Software Industry," in *8th Int. Conf. on Preservation of Digital Objects (iPRES 2011)*, Singapore, November 2011, pp. 1–9.

[6] S. Strodl, P. Petrov, and A. Rauber, "Research on digital preservation within projects co-funded by the european union in the ict programme," Vienna University of Technology, Tech. Rep., May 2011.

[7] P. Mell and T. Grance, "The nist definition of cloud computing," *National Institute of Standards and Technology*, vol. 53, no. 6, p. 50, 2009.

[8] C. Schroth, "The internet of services: Global industrialization of information intensive services," in *Digital Information Management, 2007. ICDIM '07. 2nd Int. Conf. on*, vol. 2, oct. 2007, pp. 635 –642.

[9] D. Giaretta, "The CASPAR Approach to Digital Preservation," *The Internation Journal of Digital Curation*, vol. 2, no. 1, 2007.

[10] M. A. Neumann, T. Riedel, P. Taylor, H. R. Schmidtke, and M. Beigl, "Monitoring for digital preservation of processes," in *Proceedings of the 7th international and interdisciplinary conference on Modeling and using context*. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 214–220.

[11] J. Barateiro, G. Antunes, F. Freitas, and J. Borbinha, "Designing digital preservation solutions: A risk management-based approach," *International Journal of Digital Curation*, vol. 5, no. 1, pp. 4–17, 2010.

[12] A. Avizienis, J. C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," *IEEE Trans. Dependable Secur. Comput.*, vol. 1, pp. 11–33, January 2004.

[13] M. Lankhorst, *Enterprise Architecture at Work: Modeling, Communication, and Analysis*. Springer, 2005.

[14] R. Cloutier, G. Muller, D. Verma, R. Nilchiani, E. Hole, and M. Bone, "The concept of reference architectures," *Systems Engineering*, vol. 1, no. 13, pp. 14–27, 2010.

[15] M. Jones and N. Beagrie, *Preservation Management of Digital Materials: A Handbook*. London, UK: Digital Preservation Coalition, November 2008.

[16] D. Rosenthal, T. Robertson, T. Lipkis, V. Reich, and S. Morabito, "Requirements for digital preservation systems: A bottom-up approach," *CoRR*, vol. abs/cs/0509018, 2005.

[17] J. W. Ross, P. Weill, and D. C. Robertson, *Enterprise Architecture as Strategy - Creating a Foundation for Business Execution*. Harvard Business Press, 2006.