Authenticity and Provenance in Long Term Digital Preservation: Modeling and Implementation in Preservation Aware Storage

Michael Factor, Ealan Henis, Dalit Naor, Simona Rabinovici-Cohen, Petra Reshef, Shahar Ronen,
IBM Research Lab in Haifa, Israel
and
Giovanni Michetti, Maria Guercio, **University of Urbino, Italy**

Abstract

A growing amount of digital objects is designated for long term preservation - a time scale during which technologies, formats and comunities are very likely to change. Specialized approaches, models and technologies are needed to guarantee the long-term understandability of the preserved data. Maintaining the authenticity (trustworthiness) and provenance (history of creation, ownership, accesses and changes) of the preserved objects for the long term is of great importance, since users must be confident that the objects in the changed environment are authentic. We present a novel model for managing authenticity in long term digital preservation systems and a supporting archival storage component. The model and archival storage build on OAIS, the leading standard in the area of long-term digital preservation. The preservation aware storage layer handles provenance data, and documents the relevant events. It collocates provenance data (and other metadata) together with the preserved data in a secure environment, thus enhancing the chances of their co-survival. Handling authenticity and provenance at the storage layer reduces both threats to authenticity and computation times. This work addresses core issues in long-term digital preservation in a novel and practical manner. We present an example of managing authenticity of data objects during data transformation at the storage component.[1]

1. Introduction

Long Term Digital Preservation (LTDP) is the set of processes, strategies and tools used to store and access digital data for long periods of time during which technologies, formats, hardware, software and technical communities are very likely to change. The LTDP problem includes aspects of bit preservation and logical preservation. Bit preservation is the ability to restore the bits of a data object in the presence of storage media degradation, hardware obsolescence and/or catastrophes. Logical preservation entails preserving the intellectual content of the data in the face of future technological and knowledge changes. Logical preservation is still an open research area that presents a great challenge as it needs to enable future interpretation of the preserved data by consumers that may use technologies unknown today and hold a different knowledge base from that of the data producers.

Our work leverages the Open Archival Information System (OAIS), an ISO standard for LTDP that provides a general framework for the preservation of digital assets [1]. OAIS specifies concepts, strategies and functions, and provides a high-level, flexible reference model (information and functional) for LTDP.

According to the OAIS information model, each data object requires *Representation Information* (RepInfo). The RepInfo is a set of objects used to interpret the data object. Each RepInfo may have RepInfo of its own, creating a *RepInfo network.* It ends at the knowledge base of the designated community that uses the data.

The structure preserved in the archival storage is called the *Archival Information Package* (AIP), depicted in figure 1.

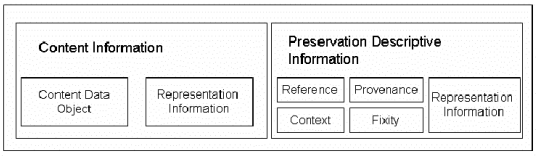

Figure 1 - AIP structure in OAIS [1]

Its content information section includes the data object and its RepInfo. In addition, the AIP contains the *Preservation Descriptive Information* (PDI) section, which includes several types of metadata: provenance guarantees the documentation of the life cycle of the AIP, fixity guarantees its integrity, context documents its relation to its environment and reference keeps a set of identifiers for the AIP.

[1] Work partially supported by European Community under the Information Society Technologies (IST) program of the 6th FP for RTD - project CASPAR contract IST-033572. The authors are solely responsible for the content of this paper. It does not represent the opinion of the European Community, and the European Community is not responsible for any use that might be made of data appearing therein.

An important part of an LTDP system is the ability to manage the authenticity of a data object. Authenticity refers to the reliability of the data in a broad sense, tracking the control over the preserved information custody. To validate authenticity of a preserved data object provenance is needed, i.e., the documented history of creation, ownership, accesses, and changes that have occurred over time for a given data object. Also a means is needed to guarantee that data is whole and uncorrupted (integrity). The maintenance of the original bit stream is not always necessary or possible; the goal is completeness of the intellectual form (meaning).

Defining and assessing authenticity are complex tasks, which include definition of roles, policies, components, and protocols for the custodial function. To enable this future assessment, a pre-requirement is the preservation of the authenticity documentation that includes the provenance data.

To address authenticity and provenance in the context of LTDP we have developed a novel model that aims to ensure the identity and integrity of a digital object. Since authenticity is not a binary attribute, our model enables evaluating the degree of authenticity. A key aspect of our approach is identifying the set of attributes that are relevant to evaluating authenticity. Another key aspect is a conceptual model to describe the dynamic profile of authenticity. This process of protecting and assessing the authenticity of a digital object is driven by an *authenticity protocol* applied to a set of digital objects. The authenticity protocol is composed of *authenticity steps,* each of which is independently executable. The authenticity steps are organized in a workflow, which defines the order of their execution. Different types of authenticity steps reference the different elements of the PDI in the AIP as defined by OAIS. A report on the evaluation results can be used (by a human or other actor) to evaluate the data authenticity. The overall model ties together these aspects along with additional components to provide an approach to managing and assessing authenticity for data subject to long term digital preservation.

Architecting and implementing a sound, consistent, and efficient LTDP system that supports authenticity and provenance is a challenge. Preservation DataStores (PDS) is our OAIS-based preservation aware storage [3,4] designed to serve as the storage component of a digital preservation system. It has built-in support for bit and logical preservation. PDS is aware of the OAIS-based preservation objects that it stores and can offload functions traditionally performed by applications. These functions include handling metadata, calculating and validating fixity, documenting provenance events, managing the RepInfo of the PDI, and validating referential integrity. Another important feature is the physical co-location of data and metadata, which ensures that metadata is not lost if raw data survives. Related AIPs are also co-located on the same media. These features allow PDS to support both bit and logical preservation, including authenticity management.

Our Preservation DataStores realizes the authenticity model by moving knowledge of provenance and other related metadata to the storage system. In particular, PDS tracks events related to ensuring the identity and integrity of the data through direct implementation of the OAIS concepts of fixity (integrity) and provenance. Further, PDS can trigger the automatic execution of the authenticity protocol (events the storage is aware of), and it supports the manual execution for external events (e.g., the change of ownership of an object). Finally, PDS ties internal changes that impact authenticity, e.g., format transformations executed via storlets, to the authenticity model, automatically making the relevant updates to the OAIS PDI. Supporting the authenticity model at the storage layer results in an optimized, robust and secure preservation environment, which can provide a stronger ability to asses the authenticity of a data object.

To summarize, our work has two main contributions:

- A novel model for managing authenticity in long term digital preservation systems
- An implementation of a preservation-aware storage system that integrates the concept of long term provenance.

The model is being implemented as part of the CASPAR OAIS framework. CASPAR [2] (Cultural Artistic and Scientific knowledge for Preservation, Access and Retrieval) is an FP7 EU project that aims to demonstrate the validity of the OAIS model with different data sets.

The rest of this paper is structured as follows. Section 2 presents related work. Section 3 presents our model and section 4 our implementation. We conclude in section 5.

2. Related Work

Among the papers most relevant to our focus on authentication and provenance we mention the provenance-aware storage system (PASS) [5] and a later work by the same group on data modeling for provenance [6]. The former [5] describes a technology that tracks the provenance of data at the file system level, and does not employ an auxiliary database for provenance management. The idea of offloading storage related activities to the storage level is similar to our PDS work. However, PDS supports also documenting provenance events external to the storage and also provenance events that are logical in nature.

In the later work [6], the authors argue that due to the common ancestry relations of provenance data, these data naturally form a directed graph. Hence, provenance data and query models should address this structure in a natural manner. A semi-structured data model with a special query language (PQL, which extends Lorel) was used, taken from the object oriented database community. Currently, PDS does not support fine-grained provenance querying.

Provenance was addressed in the context of scientific workflows [7], in terms of both the derived data and their specification. It was argued that provenance is essential for reproducibility, sharing and re-use of scientific data. The authors suggest that a workflow should systematically and automatically record provenance information for later use, replacing ad-hoc specially re-written shell scripts (e.g. perl). The proposed solution for provenance management is to provide a data capturing mechanism, a data model, and an infrastructure for ingest, access and query. The authors distinguish between prospective and retrospective provenance: the former captures the computational task, the steps (algorithm) needed to (re)create the data, whereas the latter captures the actual steps executed (as well as the execution environment setting). According to [7], provenance information is not limited to people who created/ingested/accessed the data, but may also be attributed to processes as well as recipes for data regeneration. In PDS we implement this view.

With regard to the more general storage aspects of digital preservation, previous works [8,9,10] address authentication as well as security issues. Some works [8] explore the needs of long-term storage and present a reliability model and associated strategies and architectures. Most of the previous works focus on bit preservation (maintaining bit integrity) and less on logical preservation (preserving the meaning or understandability of the data). The focus of PDS is on logical preservation.

The e-depot digital archiving system of the National Library of the Netherlands (KB) [11] is composed of the Digital Information Archiving System (DIAS). Similar to our PDS work, the e-depot library conforms to the OAIS standard, and addresses both bit and logical preservation. In DIAS, some provenance-related metadata may reside separately from the data, but in PDS, we argue for co-locating the data and metadata to improve the chances of data/meta-data co-survival – assuming that the survival of one without the other is useless.

3. Managing Authenticity

3.1 Key Concept

Authenticity is a fundamental issue for the long-term preservation of digital objects: the relevance of authenticity as a preliminary and central requirement has been thoroughly investigated by many international projects, some focused on long-term preservation of *authentic digital records* in the e-government environment, and in scientific and cultural domains[2] [12]; some devoted to the identification of criteria and responsibilities for the development of trusted digital repositories [13].

Defining and assessing authenticity are complex tasks and imply a number of theoretical and operational/technical activities. These include a clear definition of roles involved, coherent development of recommendations and policies for building trusted repositories, and precise identification of each component of the custodial function. Thus it is crucial to define the key conceptual elements that provide the foundation for such a complex framework: we need to define how, and on what basis authenticity has to be managed in the digital preservation processes in order to ensure the trustworthiness of digital objects.

One of the founding concepts for the development of a theory on authenticity is that in most cases digital objects cannot be preserved as original unchanged objects, and we only have the ability to reproduce them. Unfortunately, this runs counter to the assumption that preserving authenticity implies retaining the identity and integrity of a digital object., i.e., free from tampering or corruption. It is a sort of paradox, where preservation entails change, while authenticity needs fixity.

Authenticity cannot be recognised as given – once and forever – within a digital environment. This point implies that a clear distinction should be made between the authenticity of a preserved resource – not necessarily the original one ingested in the repository – and the procedure of *validating* that resource; the latter is a part of a more general process aimed at assuring that an information object will be kept *as an original* one i.e., reliable, trustworthy, and sound.

The authenticity of digital resources is threatened whenever they are transferred across space (i.e., whenever they are exchanged between users, systems or applications), or time (i.e., either when they are in storage or when the hardware or software used to store, process, or communicate them is updated or replaced). Therefore, the preserver's inference of the authenticity of digital resources must be supported by evidence provided in association with the resources through its *documentation,* by tracing the *history o*f its various migration and treatments, which have occurred over time. Evidence is also needed to prove

[2] The concepts presented here are rooted in the conceptual framework designed in the InterPARES project.

that they have been maintained using technologies and administrative procedures that either guarantee their continuing *identity* and *integrity* or at least minimize risks of change from the time the resources were first set aside to the point at which they are subsequently accessed.

In conclusion, authenticity is never limited to the resource itself, but is rather extended to the information/document/record system, and thus to the concept of reliability: authenticity is concerned with ongoing control over information/document/record creation process and custody. The verification of the authenticity of a resource is related to the reliability of the system/resource, and this reliability is crucially based upon complete documentation of both the creation process and the chain of preservation.

3.2 Integrity and Identity

Authenticity is established by assessing the integrity and identity of the resource.

Integrity

The integrity of a resource refers to its wholeness. A resource has *integrity* when it is complete and uncorrupted in all its essential respects. The verification process should analyse and ascertain that the essential characteristics of an object are consistent with the inevitable changes brought about by technological obsolescence**.** The maintenance of the bit flow is not always necessary or possible, but the original ability to convey meaning (e.g., maintenance of colours in a map, columns in a spreadsheet etc.) must be preserved. In other words, the physical integrity of a resource (i.e., the original bit stream) can be compromised, but the content structure and the essential components must remain the same. So, the critical issue with reference to integrity is to identify the relevant characteristics of a resource. This means understanding the nature of the resource, analysing its features, and evaluating their role so to establish what kind of changes are allowed without loss of integrity.

Identity

Identity of a resource is intended with a very wide meaning, not only its unique designation and/or identification. Identity refers to *the whole* of the characteristics of a resource that uniquely identify it and distinguish it from any other resource. In addition to its internal conceptual structure, it refers to its general context (e.g., legal, technological). From this point of view, identity is strongly related to PDI: Context, Provenance, Fixity, and Reference Information as defined in OAIS help to understand the environment of a resource. This information has to be gathered, maintained, and interpreted altogether – as much as possible – as a set of relationships defining the resource itself: a resource is not a monad with defined borders and autonomous life, it is not just a single object; a resource is an object *in the context*, it is both the object itself and the relationships that provide complete meaning to it. As a matter of fact, these relationships change over time, so we need not only to understand them and make them explicit but also to document them to have a complete history of the resource: we cannot miss it without losing a bit of the identity of the resource, with consequences on its authenticity.

3.3 Authenticity Management Tools

Authenticity management tools have to monitor and manage protocols and procedures across the custody chain to deliver the benefits of authenticity into information system, from creation to preservation.

In general, authenticity cannot be evaluated by means of a boolean flag telling us whether a resource is authentic or not. The evaluation should lead to assess the *degree* of authenticity: the certainty about authenticity is a goal and sure cases are edge cases. So mechanisms and tools for managing authenticity have to be designed keeping in mind possible alterations, corruption, lack of significant data and so on, and we need tools, mechanisms and *weights* to understand their relevance and their impact on authenticity. The consequence is that ensuring authenticity means providing a proper set of attributes related to content and context, and verifying/checking (possibly against metrics) the completeness or the alteration of this set.

Authenticity management tools have to identify mechanisms for ensuring the maintenance and verification of the authenticity in terms of identity and integrity of the digital objects by providing content and contextual information during the whole preservation process**.** The most critical issues are the right attribution of authorship, the identification of provenance in the life cycle of digital resources, the insurance of content integrity of the digital components and their relevant contextual relationships, and the provision of mechanisms to allow future users to verify the authenticity of the preserved objects or at least to provide the capability of evaluating their reliability in term of authenticity presumption. Any event producing a *change* of the object has to be described and documented at every stage in the life cycle to have, at any time, a sort of *authenticity card* for any object in the repository: the crucial point is to clearly state that the identity of an object resides not only in its internal structure and content but also – and maybe mostly –in its complex system of relationships, so that a change of the object refers not only to a change of the bits of the object but also to something around it and that anyway contributes to its identity i.e., to its authenticity.

According to the above requirements, defining a strategy for managing authenticity means:

- Identifying a set of attributes to catch relevant information for the authenticity as it can be collected along the life cycle of objects belonging to different domains
- Developing a conceptual model to describe the dynamic profile of authenticity i.e., to describe it as a process aimed at gathering, protecting, and/or evaluating information mainly about identity and integrity.

3.4 Elements of the Conceptual Model

Authenticity Protocol (AP)

The protection and assessment of the authenticity of digital objects is a process. To manage this process, we need to define the procedures to follow.

We call one of these procedures an *Authenticity Protocol* (AP). An AP is a set of interrelated steps; each called *Authenticity Step* (AS). An AP is applied to an *Object Type*, i.e., to a class of objects with uniform features for the application of an AP. Any AP may be recursively used to design other APs, as expressed by the general *Workflow* relation. See figure 2.

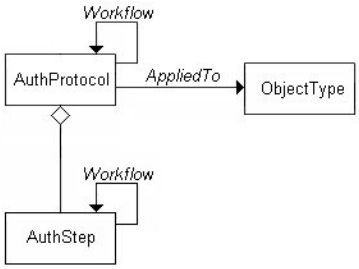

Figure 2 - Elements of the model

Authenticity Step (AS)

Every AS models a part of an AP that can be executed independently as a whole, and constitutes a significant phase of the AP from the authenticity assessment point of view. The relationships amongst the steps of an AP establish the order in which the steps must be executed in the context of an execution of the protocol. To model these relationships, we can use any workflow model, denoted as *Workflow*. An AS is performed by an *Actor Type*, a class of either human or non-human agents instantiated through the *Actor Occurrence* class. The *Actor Type* is a generalization of *Automatic Actor* and *Manual Actor* (hardware/software and human).

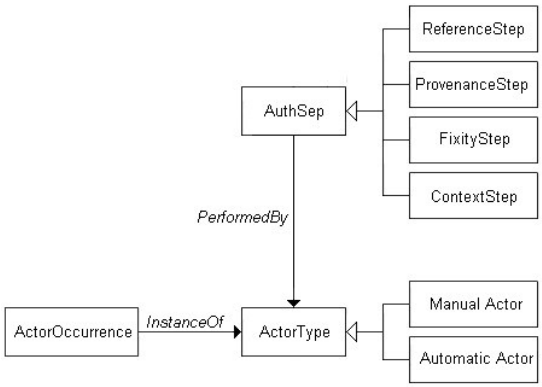

Figure 3 - Authenticity Step

There can be several types of ASs. Following OAIS, we distinguish *Step*s based on the kind of PDI required to carry out the AS. Consequently, we have four types of steps: *Reference Step*, *Provenance Step*, *Fixity Step*, *and Context Step*. See figure 3.

Since an AS involves an analysis and evaluation, we need at least information about:

- Good practices, methodologies and any kind of regulations that must be followed or can help in the analysis and evaluation
- Possibly the criteria that must be satisfied in the evaluation.

Authenticity Protocol Execution (APE)

APs are executed by an actor on objects that belong to a specific typology. The execution of an AP is modelled as an *Authenticity Protocol Execution* (APE). See figure 4.

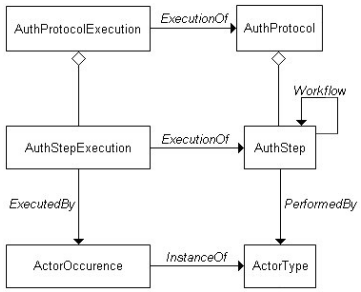

Figure 4 - Authenticity Protocol Execution

An APE is related to an AP via the *ExecutionOf* association and consists of a number of *Authenticity Step Execution*s (ASEs). Every ASE, in turn, is related to the AS via an association analogous to the *ExecutionOf* association, and contains the information about the execution, including:

- the actor who did the execution
- the information which was used

- the time, place, and context of execution.

Every ASE is executed by an *Actor Occurrence,* i.e., an instantiation of the *Actor Type*.

Authenticity Report

Different types of ASEs have different structures and the outcomes of the executions must be documented to gather information related to specific aspects of the object, e.g., title, extent, dates, and transformations. An *Authenticity Step Execution Report* simply documents that the step has been done – via the *Documented By* relation – and collects all the values associated with the data elements analysed in a specific ASE. The report provides a complete set of information upon which an entitled actor (human or application) can build a judgment, an *Authenticity Protocol Execution Evaluation*, which states an evaluation regarding the authenticity of the resource, referring to both its identity and integrity profile. See figure 5.

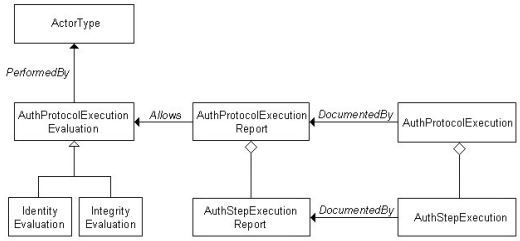

Figure 5 - Authenticity Protocol Execution Evaluation

Event

Authenticity should be monitored continuously so that any time a resource is somehow changed or a relationship is modified, an *Authenticity Protocol* can be activated and executed in order to verify the permanence of the resource's relevant features that guarantee its authenticity.

Any event impacting on a resource – and specifically on a certain type of a resource – should trigger the execution of an adequate protocol: the *Authenticity Protocol Execution* is *triggered by* an *Event Occurrence*, i.e., the instantiation of an *Event Type* that identifies any act and/or fact related to a specific *Authenticity Protocol*.

Authenticity Protocol History

The authenticity of a resource is strongly related to the criteria and procedures adopted to analyse and evaluate it: the evolution of the *Authenticity Protocols* over time should be documented – via the *Documented By* relation – in an *Authenticity Protocol History.* The evolution of an AP may concern the addition, removal or modification of any step making up the AP, and the change of the sequence defining the *Workflow*. In any case both the old and the new step and/or sequence must be retained for documentation purposes. When an AS of an AP is changed, all the executions of the AP that include an ASE related to the changed step, must be revised, and possibly a new execution is required for the new (modified) step.

See figure 6 for the overall authenticity model.

4. Authenticity and Provenance in PDS

As a preservation aware storage, PDS has built-in support for handling metadata, including provenance data. Provenance data receives special attention as it is crucial for the future usability of the content data.

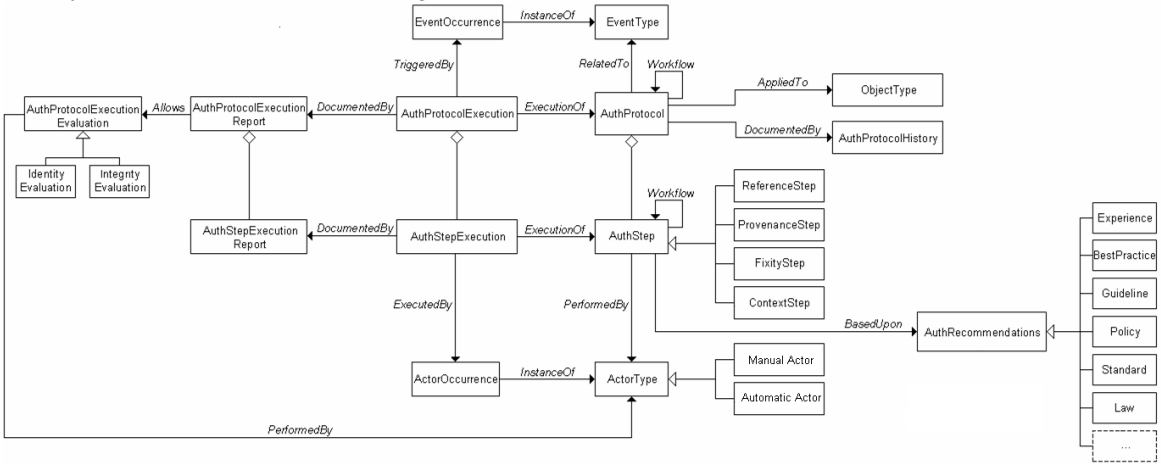

Figure 6 - Overall authenticity model

Support for handling the provenance data begins with understanding the AIP structure and having the ability to manipulate any section in it, including provenance, independently. PDS then provides additional means to support tracking the events in the life cycle of the content data and documents them as *provenance records*. This tracking is either carried out automatically in PDS when possible (if PDS is aware of the event) or triggered by an external source (e.g., human, application) by calling the designated PDS API. By tracking events and documenting them automatically in the storage we increase the likelihood that all provenance events are indeed recorded, while minimizing the dependency on updates by external management tools.

When preserving an AIP for long term, the preservation of the metadata becomes as important as the preservation of the content data and in some cases even more. Therefore, PDS preserves provenance data and other metadata along with the content data, the primary preservation target.

PDS provides a secure environment in terms of maintaining the authenticity (i.e., the identity and integrity) of the data objects by performing data intensive functions inside the storage and supporting a storlet container mechanism. Providing these features reduces the exposure of the data objects to data transfers and thus minimizes threats to the authenticity of data and metadata.

As the PDI metadata documents the relations of an AIP to its environment, PDS uses this information to co-locate related AIPs on the same media, thus providing additional support for maintaining these relations over the long term. Maintaining the relations of a data object to its environment is a core need for its authenticity management.

4.1 Structure of Provenance Data

Provenance is kept as a set of accumulative, chronologically ordered records that describe the events in the life of the content data, from its creation to date. Each record is created at a different point in time, and may have a different inner structure, depending on the knowledge and technology used in its creation.

PDS aims to satisfy two consequent issues:

1. To enable the usability and understandability of all the different provenance records by the accessing entities, which may differ substantially in identity, time and knowledge, from the creating ones and from one another.
2. To maintain a uniform provenance data set, while each record may have a different inner structure.

To address both issues each provenance record has a high level structure that contains several fields needed to manage the data set, a content field which contains the actual content of the provenance record, and a RepInfo field (see figure 7). The content of each record may have a different inner structure and the RepInfo enables its interpretation. For instance, the content may be an XML file and the RepInfo may hold its schema.

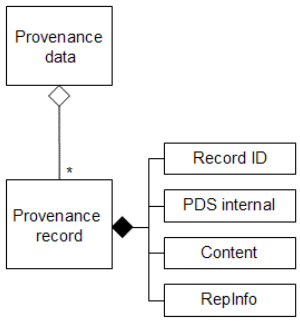

Figure 7 - Structure of provenance data

This structure provides flexibility to the provenance record through time, technology, and users, while keeping the overall understandability and uniform general appearance.

4.2 Preservation of Provenance Data

Like the content data, the provenance data also require long-term preservation to guarantee its own readability and understandability over time. If provenance data does not survive, the authenticity of the data object cannot be verified and its future use is jeopardized. Therefore, the preservation of provenance data is handled with the same level of care as the preservation of the content data itself.

To ensure the mutual survival and availability of the content data and the provenance data (along with additional metadata), the provenance is strongly encapsulated within the AIP object and all AIP sections are physically co-located on the same media (to the possible extent). When the AIP is migrated to another media or system, it is migrated as a single unit to maintain this co-location.

In addition, the usability and understandability of the records' content must also be maintained. This is done by the same preservation methods used to preserve the content data. To maintain the integrity of the metadata, fixity computations are conducted not only on the content data but also on the metadata and the results are documented in the AIP fixity section.

To maintain the understandability of the provenance records, each record is accompanied by RepInfo. As time passes and the knowledge of the user community changes, additional RepInfo may be added to each record. If the format of the provenance

data becomes obsolete, PDS enables migrating the provenance data to a new format.

4.3 Manipulating Provenance Data

The AIPs preserved in PDS may be created by different sources. In most cases, the AIP is generated externally and ingested to PDS. At ingest time, this AIP may already contain provenance data that documents the life of the content data since its creation and until its ingestion. During ingest, PDS adds a new record to document the ingest process.

In other cases the AIP, and possibly even its content data, is generated automatically inside PDS. Then, a provenance record that describes its creation, its owners, etc., is generated automatically, along with a record that documents the ingest process.

After ingest, any event in the life of the data should trigger the generation of a new provenance record to document it. Such events may be migration, transformation, access, change of ownership, etc.

When PDS is aware of an event (e.g., migration), it refers to it as an internal provenance event and records it automatically. Otherwise, if PDS is unaware of the event (e.g., change of ownership), the documentation is triggered externally and added to the provenance data by using the appropriate PDS API.

A provenance event may refer to a single AIP (e.g., creation), a group of AIPs (e.g., all AIPs that contain data in a certain format are transformed to a new format), or the entire system (e.g., the entire archive changes ownership). When a provenance event occurs, PDS may document it aside and aggregate the resulting provenance record to the appropriate AIPs later on. As AIPs may be stored offline, reading them into the system to update their provenance section can be expensive. Putting off this update until the next migration, when they are read into the system anyway, lowers the cost of this operation substantially. This optimisation is especially advantageous when a single provenance event refers to a very large set of AIPs, or even to the entire system.

In addition to recording the events in the life of the data, more complex processes related to provenance may need to be executed. (e.g., authenticity verification/validation). The PDS storlet module mechanism may be used to load and execute such process inside PDS, close to the data, minimizing data transfers and enabling optimal scheduling.

The provenance records are accumulative, meaning new records are added to the existing set of records and none of them are deleted. This mechanism enables to access the complete provenance data of an AIP as needed. Accessing provenance data may be performed independently from accessing other parts of the AIP.

4.4 Authenticity Model Support

According to the authenticity model presented above, *AuthProtocolExecution* that complies with *AuthProtocol* performs *AuthStepExecution* of different *AuthStep*s. This process is triggered by an *EventOccurence*, an instance of *EventType*; PDS may either be aware of this event or not.

If PDS is aware of the event, it automatically triggers the *AuthProtocolExecution*. The *AuthProtocol* that is used is defined in the PDS implementation and the *ActorOccurence* that executes the *AuthStepExecution*s is the PDS, as an instance of *AutomaticActor*.

If PDS is unaware of the event, it provides support to an *AuthProtocolExecution* of an *AuthProtocol* by supplying APIs for *AuthStepExecution* of different *AuthStep*s. In this case the *ActorOccurence* is an instance of *ManualActor* and may be a user or an application.

Each *AuthStepExecutionReport* is implemented as a single PDI record. A PDI record may be one of four types: provenance record, fixity record, reference record or context record. PDI records may be created automatically inside PDS, or externally by the user. In the latter case PDS supports the addition of such externally generated records to the AIP. The complete PDI section may be viewed as an *AuthProtocolExecutionReport* although it aggregates not only the documentation of a single *AuthProtocolExecution* but the documentation of all *AuthProtocolExecution*s that ever took place.

PDS is not familiar with any externally defined *AuthProtocol*s. However, if these *AuthProtocol*s are preserved as AIPs in the system, PDS tracks the *AuthProtocolHistory* transparently as part of their preservation process.

AuthProtocolExecutionEvaluation may be carried out externally, getting its input by accessing the relevant PDI records, or internally, using a storlet module loaded to PDS. Specifically for *IntegrityEvaluation*, PDS supplies a dedicated API that validates the integrity of an AIP.

4.5 Use Case: Transformation

Suppose that long-term preservation of the digital file containing the present paper is required. The file would be ingested to PDS as an MS-Word document, along with RepInfo describing the format and PDI describing the creation of the document and the relationships with its environment. Years go by, until it is decided to transform the preserved DOC file to PDF format.

The EventOccurence is data transformation from DOC to PDF, executed in PDS. The result of the transformation, the data in PDF format, needs to be preserved in a new AIP with new RepInfo (e.g., PDF specification) and related PDI. This new AIP is ingested to PDS; the new RepInfo, if does not already reside in PDS, is also ingested as a separate AIP. For the sake of simplicity, assume this RepInfo already exists as an AIP in PDS.

The AuthProtocol related to this EventType is implemented in PDS and includes the following AuthSteps:

1. ProvenanceStep: document the transformation in the original AIP.
2. ProvenanceStep: document the creation procedure, owner, etc., of the new AIP.
3. ReferenceStep: generate a unique identifier for the new AIP; this identifier is a version of its originator's identifier.
4. FixityStep: compute fixity on the new AIP.
5. ContextStep: describe the relation of the new AIP to the original AIP.
6. ProvenanceStep: document the ingest of the new AIP.

PDS is an instance of AutomaticActor. It performs the AuthStepExecution of the AuthSteps detailed by the AuthProtocol. For each step it generates a PDI record that serves as an AuthStepExecutionReport. These PDI records are preserved in the AIPs and may be referred to as the AuthProtocolExecutionReport.

5. Conclusions and Future Work

5.1 Conclusions

Preservation of digital objects is critically dependent on successful preservation of their authenticity. Authenticity is not only a factor of a successful preservation; it is a requirement, a necessary condition without which a failure of the preservation system is implied. Users must be confident that the objects managed by system are the original ones, or at least must have confidence that they are "like" the original ones, or that in the worst case, they can be traced in some auditable way to the original objects. Authenticity determines the users' attitude towards the preserved objects in terms of reliability and trustworthiness.

Assessing authenticity means evaluating integrity and identity, with special reference to the context and provenance. Given the inevitable changes that digital objects and their environments undergo, to maintain them over time across different hardware, software, and changes of custodians, it is crucial to understand what identity is, to what extent we can modify objects and environments while preserving identity, and which are their essential properties. Provenance information is key to building a sound preservation strategy and system by which creators, custodians and owners are traced together with the general context, and the identity of the stored objects is preserved.

The authenticity model presented here has been designed to meet the requirements and objectives of an authenticity-aware preservation. The model is compliant with OAIS ISO 14721:2003 [1] and can serve as a basis for further refinement. It aims to clarify vague and/or overlapping concepts, and defines a small set of classes and associations that can be implemented flexibly. Like other models [14], it is based on the notion of process as an activity that somehow changes an object. Unlike other models, it does not focus on provenance per se, but rather interprets it as a fundamental key to the more general notion of authenticity. Like the OPM model [14], it allows describing the chain of actions performed on an object. Unlike other models, it does not finish with this descriptive perspective, but it relies on it to depict the evaluation stage providing the authenticity judgement.

Preservation aware storage follows current trends of offloading down to the storage layer functionality traditionally performed by applications. Among these functions, PDS supports maintenance and evaluation of authenticity for the preserved digital objects.

The implementation of PDS largely supports the authenticity processes as modelled here. It provides an automatic actor for internally coded authenticity protocols and supports manual authenticity steps execution via APIs and a "storlet" mechanism. Both internal and manual actors' execution steps are recorded and preserved as part of the PDI section of AIPs (for the different parts of an AIP, see the OAIS model [1]). As a result, future evaluations of the object's identity and integrity are enabled.

Offloading high-level functions to the (lower level) storage reduces network traffic and provides a more secure environment for the data objects. PDS co-locates the sections of each AIP and tries to group related AIPs on the same media. These features help protect the authenticity of the data objects and their relations to their environment at the system level.

Since authenticity is an ongoing process that involves many AIPs, the cost of its execution may be very high. PDS optimises the scheduling of these operations thus making the process feasible.

By providing authenticity support at the storage layer, PDS takes the concept of preservation aware storage to a new level. The authenticity model and the PDS implementation presented here address core issues in long-term digital preservation in a novel and practical approach.

5.2 Future Directions

One major research area is related to the design of an authenticity evaluation system. In the long term lifecycle of a digital object it is likely that alterations (e.g., change in format) will occur. Hence, an evaluation framework is needed to understand the relevance and impact of the changes on authenticity.

One possibility is to construct a probabilistic confidence model to assess the degree of an object's authenticity. Such a model can take into account the level of control we have both over the object and the processes with which it was involved, as well as the quality of provenance events documentation. Users may be provided with an *authenticity probability* on which they can base their own authenticity evaluation. Re-thinking authenticity with a less deterministic approach, the authenticity evaluation model may use weights and metrics in conjunction with recorded events and/or processes to produce a probabilistic confidence level, rather than a forced binary choice.

Future investigation should also be devoted to a refinement of the model, based on feedback from practical experience in specific areas. Further effort should be invested to integrate some important outcomes from other projects such as PREMIS [15] into the model. The evaluation of other approaches and theoretical results is an ongoing research activity [14].

Finally, an ontological representation of the model through ISO 21127:2006 should be the next step towards formalizing and sharing concepts in a broad environment.

5.3 Enhancements to the PDS Implementation

To enhance and complete the support for the authenticity model, PDS should enable authenticity protocols explicitly, e.g., by implementing AuthProtocol as an object and preserving each protocol as an AIP. PDS should support the execution of such a protocol object that specifies the steps to execute. By doing so, the pre-defined protocols implemented in PDS may also be documented and preserved. In addition, external users could access the protocols and view their content in a human understandable manner. PDS API should be extended to support loading and executing authenticity protocols; then a complete protocol execution may be triggered by a single API call. This enhancement will treat equally internal (automatic) and external (manual) protocol executions. The authenticity protocol history will be documented transparently for all protocols by preserving them as AIPs in the system.

6. References

[1] ISO 14721:2003. Space data and information transfer systems – Open archival information system – Reference model. http://public.ccsds.org/publications/archive/650x0b1.pdf.

[2] CASPAR web site; http://www.casparpreserves.eu/

[3] M. Factor, D. Naor, S. Rabinovici-Cohen, L. Ramati, P. Reshef, and J. Satran. "The Need for Preservation Aware Storage – A Position Paper". *ACM SIGOPS Operating Systems Review*. Special Issue on File and Storage Systems, Vol. 41, Issue 1 (Jan 2007), pages 19-23.

[4] M. Factor, D. Naor, S. Rabinovici-Cohen, L. Ramati, P. Reshef, J. Satran, D.L. Giaretta. Preservation DataStores: Architecture for Preservation Aware Storage. MSST 2007: 3-15.

[5] K. Muniswamy-Reddy, D.A. Holland, U. Braun, and M. Seltzer. "Provenance-aware storage systems". In *Proceedings of the 2006 USENIX Annual Technical Conference*. June 2006.

[6] D. A. Holland, U. Braun, D. Maclean, K.K. Muniswamy-Reddy, and M. Seltzer. *Choosing a Data Model and Query Language for Provenance.* In proceedings of the 2nd International Provenance and Annotation Workshop, Salt Lake City, UT, Jun 2008.

[7] S.B. Davidson and J. Freire. "Provenance and Scientific workflows: Challenges and Opportunities", SIGMOD 08 june 9-12, 2008, Vancouver, BC, Canada

[8] M. Baker, K. Keeton, and S. Martin. "Why traditional storage systems don't help us save stuff forever". Technical Report 2005-120, HP Laboratories Palo Alto, June 2005.

[9] M. Baker, M. Shah, D. Rosenthal, M. Roussopoulos, P. Maniatis, T.J. Giuli, and P. Bungale. "A fresh look at the reliability of long-term digital storage". In *Proc. European Systems Conference (EuroSys),* April 2006.

[10] M. W. Storer, K.M. Greenan, E.L. Miller, K. Voruganti. "POTSHARDS: Secure Long-Term Storage Without Encryption". In *Proceedings of the 2007 USENIX Technical Conference*, June 2007.

[11] Erik Oltmans, Raymond J. van Diessen, Hilde van Wijngaarden. "Preservation Functionality in a Digital Archive". *Digital Libraries, 2004 ACM/IEEE Joint Conference on (JCDL'04*), 2004, pp. 279-286. See http://www.kb.nl/e-depot. DIAS – Digital Information Archiving System. See http://www-5.ibm.com/nl/dias/preservation.html.

[12] InterPARES – International Research on Permanent Authentic Records in Electronic Systems. See http://www.interpares.org/.

[13] Research Libraries Group. Trusted Digital Repositories: Attributes and Responsibilities. Mountain View (CA), 2002; NESTOR Project http://www.langzeitarchivierung.de/index.php.

[14] L. Moreau, J. Freire, J. Futrelle, R.E. McGrath, J. Myers, P. Paulson: The Open Provenance Model: An Overview. IPAW 2008: 323-326, Salt Lake City, UT, USA

[15] PREMIS – PREservation Metadata Implementation Strategies. http://www.loc.gov/standards/premis/v2/premis-2-0.pdf

[16] CASPAR – Cultural, Artistic and Scientific knowledge for Preservation, Access and Retrieval. See http://www.casparpreserves.eu/.

[17] ISO 21127:2006. Information and documentation – A reference ontology for the interchange of cultural heritage information.

[18] NARA – National Archives and Records Administration. See http://www.archives.gov/.

[19] D. Giaretta et al. "Supporting e-Research Using Representation Information". Paper read at Proceedings of the UK e-Science All Hands Meeting, 19th - 22nd September, at Nottingham UK, 2005.