https://doi.org/10.2352/issn.2168-3204.2019.1.0.11

Preservation of Evolving Complex Information Objects

Ivan Subotic, Lukas Rosenthaler; Digital Humanities Lab, University of Basel; Basel, Switzerland

Abstract

Having trustworthiness as the driver, the long-term preservation of evolving complex information objects from a RDF-based Virtual Research Environment (VRE) has to ensure the integrity, authenticity, and provenance of the research data it encompasses. Besides the known difficulties, preservation of evolving complex information objects from a VRE provide additional challenges, as not only the objects created inside the VRE but also the VRE as such with its ontologies describing the structure of the digital objects, and additionally any referenced bitstream data, can evolve and change over time. This change over time needs to be captured in such a way, so that not only each object can be recreated to any version from its past, but also its context, namely all surrounding and connected digital objects, and correspondingly also their context and so on. Further, we propose to store all fixity information of the digital objects themselves and also of the provenance to a public blockchain, where it would serve as a single source of truth, which all users could trust.

Introduction

At the DaSCH[1], our main goals are the creation of a trusted repository for research data in the Humanities, their long-term usability and preservation. The DaSCH repository is build on Knora [1], a RDF[2]-based Virtual Research Environment (VRE), providing the following feature set (only an excerpt):

- the storing of different project specific data-models in the same Triplestore,
- write access to the data, allowing projects to not only publish their data but also create their data on the same platform,
- to promote re-use, provide the possibility for interlinking data between projects, in the future even across repository boundaries, and
- to allow citing, provide versioning of the data.

As a result of the mentioned features, the content of the repository can be seen as evolving complex information objects. As complex information objects we understand objects that are comprised of or are part of other information objects. In their simplest form, they are defined solely in RDF, e.g., a Person with all its properties such as name, address, etc. Additionally, objects can also be defined in RDF and include references to bitstream data, e.g., a Book, where the properties of the book like title, author (a link to a Person), publisher, etc. are defined in RDF, and each page additionally references an image of the digitized physical page.

Having trustworthiness as our driver, a solution for long-term preservation of evolving complex information objects has to ensure the integrity, authenticity, and provenance of the research data it encompasses. The solution should further be ISO 14721:2012 [8] OAIS Reference Model compliant and move us in the direction of achieving ISO 16363:2012 [9] Audit and Certification of Trustworthy Digital Repositories.

Besides the known problems [2, 6] of authenticity, provenance, and integrity, to just name a few, the preservation of complex information objects from a RDF-based VRE provide additional challenges. In such an VRE, not only the data objects created inside the VRE (people, books, etc.) but also the data structure itself can evolve and change over time. The data structures and/or semantic definitions are described through ontologies, with such definitions at the system level, i.e. the *system entity definitions* and at the project level, i.e. *project entity definitions*. As an example, the afore mentioned project specific Person class and its properties can be a subclass and sub-properties of some system level entities.

In a broad sense, we can express the reliability of the preserved data through authenticity. To be able to validate authenticity, we need provenance, which can be described as the complete documented history of the digital object's life, from creation including ownership, accesses, and any changes or transformations that have occurred over time. Thus, to provide provenance, any changes to the complex information objects that happen over time need to be captured in such a way, so that not only it is captured for each digital object, but also its context, namely all surrounding and connected digital objects, and correspondingly also their context and so on.

Fixity information takes an important role in the preservation of complex information objects and more so in an automated system. At any detected loss of integrity, counter measures need to be automatically launched as to restore the integrity of the information object. If the information object cannot be repaired solely by the information it carries itself, other remote replicas need to be used [6].

Additionally, in the world of Linked Open Research Data, fixity information needs to be made public, so that it can serve as a means for checking the authenticity and provenance of the accessed digital objects. Further, fixity information needs to be stored in an unchangeable way, as to not allow any malicious or inadvertent changes. For this reasons, we propose to store all fixity information of the digital objects themselves and also of the provenance to a public blockchain. A blockchain is inherently auditable, unchangeable, and open, and would provide public access to fixity information, where it would serve as a single source of truth, which all users could trust.

The main contribution of this paper is twofold: first, we outlines the challenges in long-term preservation of RDF-based evolving complex information objects, and second, describe one possible solution.

Related Work

Fedora Commons and DSpace [12], are both open source digital repositories for managing (complex) digital objects. They have organizationally merged into DuraSpace[3].

The Repository of Authentic Digital Records (RODA) [4] is

[1] Swiss Data and Service Center for the Humanities, http://dasch.swiss

[2] RDF – Resource Description Framework

[3] http://duraspace.org

[4] https://demo.roda-community.org

an open source digital archival repository. It is built on Fedora and can support the existing XML metadata schemas, such as the Encoded Archival Description (EAD) [3], the Metadata Encoding and Transmission Standard (METS) [4] and the Preservation Metadata: Implementation Strategies (PREMIS) [5]. In terms of preservation actions, the repository supports normalisation in ingest and other actions such as format conversion and checksum verification.

The group of digital repositories based around solutions using Fedora or DSpace could certainly be integrated as to provide long-term preservation services for the type of information objects we have. We decided against this route, because there is a certain feature overlap with Knora, so that these repositories bring more than we need to the table.

SHAMAN[5] uses iRODS [11], an integrated rule-oriented data Grid as implementation technology. iRODS provides transparent support for local and remote storage and even allows to additionally leverage the Cloud [10] for storage purposes. It provides a distributed preservation-policy and workflow-driven preservation environment, with a strong focus on preservation of context, discoverability of the content, and risk management through geographically dispersed replication support.

Our chosen approach is based on iRODS. iRODS allowing us to integrate well with Knora through custom extensions, while all necessary preservation actions are provided by iRODS. Further, iRODS has connectors for Fedora and DSpace, allowing us to also pursue these avenues if at a later point in time it is deemed necessary.

Complex Information Objects

Earlier, we have given a short definition of the term Complex Information Object, upon which we will expand in this section.

As an example, we will look at two projects, Project A and Project B. Project A is a collection of digitized books, while Project B is a collection of people.

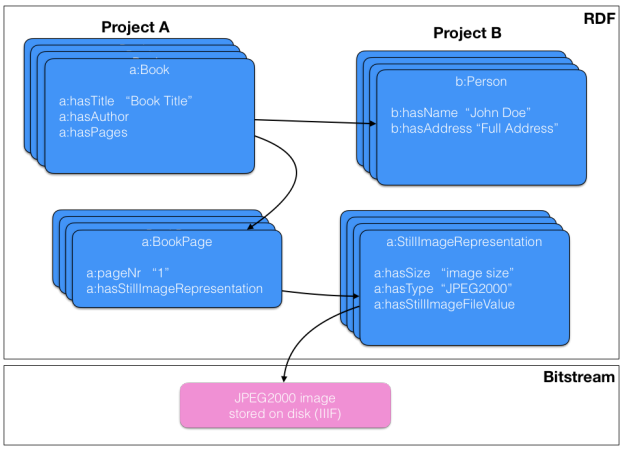

Figure 1. *Complex Information Objects.*

Figure 1 depicts one complex information object from each project. We see that the book from Project A has three properties. The first property `a:hasTitle` simply points to a string representing the book's title. The properties `a:hasAuthor` and `a:hasStillImageRepresentation` are a bit different, as they point to other objects. The `a:hasAuthor` property points to the Project B's object representing a person, conveying the meaning that this person is the author of the book. The `a:hasPages` property points to a series of book's page objects. Each page then points through the `a:hasStillImageRepresentation` property to `StillImageRepresentation` objects, which in turn point to the image's bitstream on disk.

The structure and semantics of each class of objects like our `a:Book` in the example and its properties like `a:hasTitle` are defined in ontologies expressed in RDF.

These are just two simplified examples, representing the two categories of complex information objects, namely the ones represented only through RDF and the ones represented through RDF in combination with bitstream data.

Additionally, each property value is versioned (not depicted) and can change over time. The versioning information allows querying each complex information object and its surrounding context at specific points in time, thus allowing direct citing. Also, we briefly mentioned in the introduction, that in the future it will be possible to link complex information objects residing in different repositories, which adds another layer of complexity which needs to be taken into account by the Long-Term Preservation Layer.

The term Information Object is also defined as part of the OAIS [8] Information Model, where at the highest level of abstraction, it is represented as a composition of a Data Object and Representation Information necessary for full interpretation. The Representation Information is further composed of Structure Information, Semantic Information and Other Representation Information. The Reference Model further distinguishes different types of Information Objects such as Content Information, Preservation Description Information, Description Information, etc.

In the terms of the Reference Model, our above example of the book is a Content Data Object and the ontology describing its structure and semantics are the Representation Information, forming together Content Information.

Challenges in Tracking Provenance

Earlier we have introduced that in an RDF-based VRE, not only the data objects created inside the VRE (people, books, etc.) but also the ontologies, i.e. the data structure and/or semantic definition itself can evolve and change over time, which needs to be documented. We can see ontologies as metadata to the RDF data, and thus provenance information as meta-metadata.

The changes that need to be documented can be grouped into three broad categories: Ontology Changes, RDF Data Changes, and Bitstream Changes.

Ontology Changes: These are any changes to the ontology, i.e. the underlying data model with which the structure and/or semantic definitions of the RDF data objects is described. For example, adding a new property like alternative address to the Author class. These ontology changes can happen on the system or project level. Also they can be performed inside the application, when initiated through the user interaction, or offline, when performed outside of the application. The online changes can be tracked and automatically provided to the preservation layer, while the offline changes need to be manually provided to the preservation layer. These are some examples of such changes:

1. Changes to the definition of an ontology entity (class or property), e.g., revision of the text that defines the meaning of the entity, revision off relationships to other related entities, etc.

[5]https://cordis.europa.eu/project/rcn/85468/factsheet/en

2. Changes to the ontology entity itself, e.g., renaming of classes or properties, addition/deletion of properties to a class, etc.
3. Changes of structural constraints of an ontology entity, e.g., revision off value type constraints of a property, revision off mandatory revels of a property, revision off iteration rules of a property, etc.

RDF Data Changes: These are changes to the RDF data, e.g., adding a new value to a property like the alternative address we've added before. These are usually changes which are happening through normal usage of the VRE through user interaction (online), but can also stem from offline (outside) changes:

1. Online (inside the application) changes to the data. These changes can be tracked by the VRE and provided to the preservation layer.
2. Offline (outside of the application) changes to the data. These changes could be necessitated through system or project ontology changes, which would leave the data inconsistent. These changes cannot be tracked by the VRE and need to be provided manually to or detected automatically by the preservation layer.

Bitstream Changes: These are changes to the bitstream data. After a bitstream is ingested, no changes are allowed. Only new versions can be generated through some form of transformation, usually for data format migration purposes.

Proposed Solution

Figure 2 depicts the architecture of the proposed solution in broad strokes. Here we see the three distinct layers, namely the Virtual Research Environment (Knora), the Long-Term Preservation Layer (iRODS + Extensions), and the public Blockchain (Ethereum).

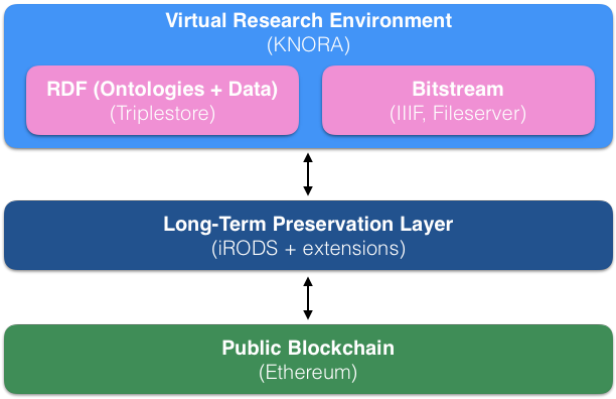

Figure 2. *Architecture of the proposed solution.*

Long-Term Preservation Layer

Why did we decide to add a Long-Term Preservation (LTP) Layer and not extend the data-model of Knora to support long-term preservation? Our first designs where based around extending the data-model of Knora to support long-term preservation, like adding checksums. It quickly became apparent, that this approach falls apart, as soon es there are changes to the content which cannot be tracked back, like some forms of Ontology Changes, which require changes to the data, to keep everything consistent. The moment such changes are performed on the data, the stored checksums become invalid. Also, since know the stored ontologies and the content data where changed in place, the previous state cannot be reproduced anymore. These leads to discontinuity in provenance, which is not a good thing in a LTP-System. Another, also important reason, is the long-term nature of preservation, i.e. that the preservation of the data needs to be continued even in the event that the Knora VRE layer is not. Further, keeping a strict separation, between Knora as the producer, and the LTP-Layer as the archive, allows to exchange the layers for different implementations in the future.

The scope of the LTP-Layer covers the OAIS functional entities of Ingest, Archival Storage, and Access.

Information Packages

In the context of the OAIS Reference Model [8] we define Knora [1] as the OAIS Producer sending Submission Information Packages (SIPs) to the Long-Term Preservation (LTP) Layer. Further, Knora should be able to consume the Dissemination Information Packages (DIPs) produced by the LTP-Layer.

The Information Packages will follow the specifications of the E-ARK family of specifications maintained by the Digital Information LifeCycle Interoperability Standards Board (DILCIS Board)[6]. These specifications include the E-ARK Common Specification [13], the E-ARK SIP Specification [14], the E-ARK AIP Specification [15], and the E-ARK DIP Specification [16]. Having Interoperability as the goal, the Common Specification for Information Packages allows to be used with content of any type or format. This is achieved by introducing the concept of Content Information Type Specification. By describing in detail the requirements for handling of the content, metadata, and documentation for specific document types, in our case the Knora Complex Information Object, the Content Information Type Specification can be used to extend and customize the Common Specification for Information Packages.

For the description of preservation information we propose to use the PREMIS Data Dictionary [5]. It is a widely used international metadata standard for the preservation of digital objects. It specifically supports the long-term preservation process with the explicit goal of ensuring integrity, authenticity, provenance, readability and usability. PREMIS also emphasises the documentation of the objects provenance and relationships among different objects, where the current version 3.0 expands the scope beyond repository boundaries. This is a feature which will become important in the future, when the Knora VRE adds the support for interlinking Complex Information Objects residing in different repositories.

The PREMIS Data Model defines four Entities, which are Objects, Events, Agents, and Rights. The documentation about the actions, is aggregated as an Event. Thus, Events are a crucial component for provenance description associated with Objects. PREMIS OWL ontology defines classes and properties which allows us to describe the preservation metadata in RDF.

PREMIS requires that any changes to data are documented in the form of change events. Based on the described ontology, data, and bitstream changes from the previous section, we can define three groups of change events, namely *Ontology Change Events*, *Data Change Events*, and *Bitstream Change Events*.

The preservation system will detect and alert regarding these changes, but manual interaction will be necessary, to provide a detailed description for the reasons of the change even, e.g., update to a new version of the system ontologies, which required changes to the data.

For each *Change Event*, the involved complex information

[6] http://www.dilcis.eu

objects will be packaged in SIPs and sent to the LTP-Layer for ingestion. The LTP-Layer will periodically query Knora as to detect possible offline changes.

iRODS

The core of the Long-Term Preservation Layer will be implemented through iRODS the integrated Rule-Oriented Data System, an open source data management software. It allows data management through predefined rules on virtualized data storage resources, allowing the underlying storage services to expand as the need arises.

In iRODS, files are stored as Data Objects, which are organized into Collections inside a virtual filesystem. The Data Objects correspond to files and the Collections to subdirectories, with some distinctions. The Collections don't have any reference to the actual physical storage path, allowing two Data Objects to be stored in different physical locations. A single Data Object can refer to multiple replicas of the underlying file stored in different locations and/or storage technologies. Data Objects and Collections are stored in Storage Resources. They provide a layer of abstraction between the Data Objects and the actual physical storage location, allowing to be configured to use different storage technologies and perform automated replication.

Preservation policies can be implemented in iRODS through the use of microservices. Besides the already provided microservices, iRODS allows to be extended through the creation of custom microservices. This way we can implement the necessary preservation functionalities corresponding to the OAIS functional entities of Ingest, Data Management, Archival Storage, and Access.

Blockchain

In the context of long-term preservation, fixity information plays a very important role, since it is our shield against unauthorized changes to the data in our custody. Storing this important piece of information together with the data, which it is used to shield, although convenient, allows potential malicious changes to be performed undetected, since while changing the data, also the fixity information could be changed.

To provide transparency and further the trust in the repository, we can publish key peaces of preservation information, like fixity and main change events to a public Blockchain like Ethereum[7]. A public blockchain would provide a highly distributed and available read-only storage, which is inherently auditable, unchangeable, and open, where it would serve as a single source of truth, which all users could trust. This information can then be used to additionally audit and validate the information entrusted to the long-term preservation system.

Conclusion and Future Work

In this paper we have outlined additional challenges posed by data stemming from a RDF-based Virtual Research Environment. We proposed a solution for preservation of such evolving complex information objects.

Future work will entail raising the scalability of the solution. For better scalability, we need to raise the granularity preserved for each new version of a complex information object, especially the context, as the current solution entails storing a large portion of the data for each new version of an information object. Further, the detection of offline changes to the data in Knora needs to be optimized, so that the performance will keep up with the anticipated growth of data stored inside the VRE.

[7] https://www.ethereum.org

References

[1] KNORA: Knowledge Organization, Representation, and Annotation, Available at: `http://www.knora.org` [Accessed March 15, 2019].

[2] B. Houghton, Preservation challenges in the digital age, D-Lib Mag., vol. 22, no. 7–8, Jul. 2016.

[3] The Library of Congress: Encoded archival description: official site, `https://www.loc.gov/ead/` [Accessed March 12, 2019].

[4] The Library of Congress: Metadata encoding and transmission standard, Available at: `http://www.loc.gov/standards/mets/` [Accessed March 12, 2019]

[5] PREMIS Editorial Committee, PREMIS Data Dictionary for Preservation Metadata, version 3.0, June 2015.

[6] Ivan Subotic, Lukas Rosenthaler and Heiko Schuldt, A Distributed Archival Network for Process-Oriented Autonomic Long-Term Digital Preservation, Proceedings of the 13th ACM/IEEE Joint Conference on Digital Libraries (JCDL '13), Indianapolis, IN, USA 2013/7

[7] Ivan Subotic, A Distributed Archival Network for Process-Oriented Autonomic Long-Term Digital Preservation, PhD Thesis, Department of Mathematics and Computer Science, University of Basel, Switzerland 2013/4

[8] ISO 14721:2012, Space Data and Information Transfer System, Open Archival Information System (OAIS) – Reference model (2012).

[9] ISO 16363:2012, Space Data and Information Transfer Systems, Audit and Certification of Trustworthy Digital Repositories (2012).

[10] P. Wittek and S. Darányi, Digital Preservation in Grids and Clouds: A Middleware Approach, *Journal of Grid Computing*, 10(1):133–149, Mar. 2012.

[11] A. Rajasekar, R. Moore, C.-Y. Hou, C. A. Lee, R. Marciano, A. de Torcy, M. Wan, W. Schroeder, S.-Y. Chen, L. Gilbert, P. Tooby, and B. Zhu, iRODS Primer: Integrated Rule-Oriented Data System, 2010.

[12] R. Tansley, M. Bass, D. Stuve, and M. Branschofsky, The DSpace Institutional Digital Repository System: Current Functionality, In Proc. JCDL'03, pages 87–97, 2003.

[13] DILCIS Board, E-ARK Common Specification, Available at: `http://dilcis.eu/specifications/common-specification` [Accessed March 15, 2019].

[14] DILCIS Board, E-ARK SIP Specification, Available at: `http://dilcis.eu/specifications/sip` [Accessed March 15, 2019].

[15] DILCIS Board, E-ARK AIP Specification, Available at: `http://dilcis.eu/specifications/aip` [Accessed March 15, 2019].

[16] DILCIS Board, E-ARK DIP Specification, Available at: `http://dilcis.eu/specifications/dip` [Accessed March 15, 2019].

Author Biography

Ivan Subotic received his PhD in computer science from the University of Basel (2013) since then he has worked at the Digital Humanities Lab, University of Basel and the Data and Science Center for the Humanities, where he is working on distributed long-term digital preservation (P2P, GRID, cloud), long-term access, semantic archives, and semantic web technologies.

Lukas Rosenthaler studied physics and astronomy in Basel and received his PhD also there. He worked as a Postdoc at ETH Zurich. He wrote his habilitation in the humanities department of the university of Basel about long-term archiving of digital data. Since 2012, he's the head of the Digital Humanities Lab and since 2017 he's the director of the Data and Service Center for the Humanities DaSCH.