Towards a Framework for Managing Evolving Information Resources on the Data Web

Marios Meimaris[*], George Papastefanatos[+], Christos Pateritsas[*], Theodora Galani[*], Yannis Stavrakas[+]

Institute for the Management of Information Systems, Research Center "Athena", Greece
{m.meimaris, gpapas, pater, theodora, yannis}@imis.athena-innovation.gr

Abstract. The web of data has brought forth the need to preserve evolving information within linked datasets; however, a basic requirement of data preservation is the maintenance of the datasets' structural aspects as well. In this paper, we present a linked data approach for the preservation and archiving of open heterogeneous datasets that evolve through time, at both the structural and the semantic layer, taking into consideration the requirements for modelling evolving linked datasets.

Keywords: Data Evolution, Change Management, Linked Data Dynamics

1 Introduction

The Data Web has brought forth a need to treat the web as a dynamic accumulation of facts created within collaborative environments that can be processed and combined in order to extract new knowledge. The benefits of evolution management in this context are placed into two categories, (i) quality control and maintenance and (ii) data exploitation. Evolution management addresses the following challenges: dataset synchronization, link maintenance, schema and entity evolution and versioning [12]. Data-aware practices make persistence, accessibility and usability value adding attributes [1][2]. In this paper, we advocate the need for addressing the problem in multiple dimensions through a framework that combines versioning, provenance, change detection and quality control. As a basis, a conceptual model that supports the representation of constructs relevant to the aforementioned dimensions and treats simple as well as complex changes as first-class citizens is presented herein.

Section 2 includes relevant work, section 3 discusses the requirements for evolving LOD datasets, section 4 presents the evolution space model and its components, and section 5 provides a conclusion and discusses future directions.

[*] Work supported by the EU project DIACHRON

[+] Work supported by the EU/Greece funded KRIPIS: MEDA Project

2 Related Work

In [3][4] the authors extend HTTP with a temporal dimension for accessing past versions of web documents and LD resources. In [5], they provide extended versioning functionality to the web server. In [6] the authors address multi-versioning for XML documents by using deltas between sequential versions. [7] proposes a method for archiving scientific data from XML documents, by targeting individual elements in the tree and pushing down time to the children nodes in order to assert changes, an approach also followed in [8]. [9] differentiates between the document-centric and entity-centric perspectives of LOD change dynamics, a distinction we partially adopt, as will be described further on. [10] computes the semantic and structural differences between versions of a RDFS graph. [11] deals with dataset dynamics in distributed LD, and identifies several levels for the requirements of change dynamics: (1) vocabularies for describing dynamics and representing changes, (2) protocols for change propagation and (3) algorithms for change detection. They implement a change detection framework which incorporates these points in a unified functionality scheme.

3 Requirements for evolving information resources

Most of the challenges in LD dynamics stem from the decentralized nature of publishing and curating interdependent datasets across disparate sites. In contrast with traditional settings where evolution is performed within a controlled and monitored environment, the Data Web poses new requirements for dataset evolution dynamics:

Persistent identification and reference. An Identifier mechanism is needed that will reify the id information, e.g., primary keys must be converted to persistent citable URIs. Representations must capture both temporal and time-agnostic characteristics. Thus, the identifiers must be able to abstract from time.

Simple and Complex Changes. Changes can be asserted in a multitude of levels, depending on the semantic richness. In [13] there is a hierarchical differentiation of changes that considers additions and deletions as the building blocks for higher-level changes. A formal hierarchical representation model is required and it must be possible to define complex changes on higher semantic structures [8].

Temporal and provenance annotations. Provenance management enables trust, interoperability and licensing, by capturing dataset lineage. It can be modelled in many granularities, from datasets to individual facts. We consider time to be part of provenance, making temporal provenance a direct enabler of dataset versioning and evolution. We adopt the partitioning of time into transaction-time and valid-time.

Common abstraction model. LOD use heterogeneous data models, including standard and/or ad hoc or proprietary formats. Diachronic preservation should exhibit format-independence, data traceability and reproducibility and an overall common denomination for data that originate from different models.

Support for low-level and high-level preservation. The model must be able to capture the evolution of both the structural aspects of datasets and the evolution of information entities across time.

Multi-versioning and longitudinal querying. The framework must answer several types of queries, within a version or across sets of versions. It should support dataset listing, complete/partial retrieval, longitudinal queries and change queries.

4 Modelling Evolution: the 2x2 Model Space

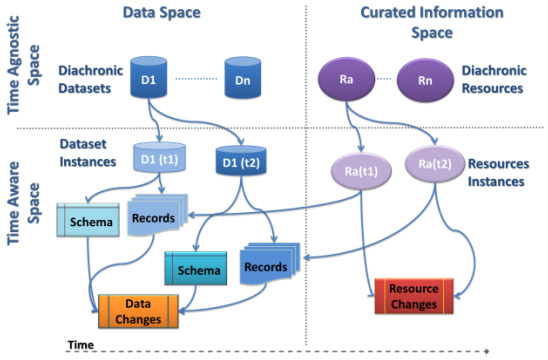

Fig. 1. The 2x2 Model Space

The model space is comprised of the following two dimensions: time awareness and semantic awareness. At the core lies the notion of the evolving entity as an abstraction of all entities. Evolving entities are identifiable and citable objects. The main entity types are:

- **(i)** Datasets: conceptual entities that represent a particular dataset from a time-agnostic point of view (diachronic datasets) and a time-aware point of view (dataset instantiations).
- **(ii)** Schema objects: these represent the schema-related entities of the archived datasets given the dataset's source model.
- **(iii)** Data Objects**:** these consist of records and record attributes. A record represents a data entry about a particular evolving entity. Records are uniquely identified in order to allow record-level annotations.
- **(iv)** Diachronic Resources: these are concepts defined through a declarative way over a dataset. They provide a curation mechanism to define contexts of evolution and relate high level changes to them.
- **(v)** Record, Schema and Resource Sets: These are collections of their corresponding entities that exist within a particular dataset instantiation. This way they become pluggable and interchangeable across versions.
- **(vi)** Change: these come in Change Sets between two instantiations of the same dataset. When applied to collections of entities, change sets are specialized to record set, schema set and resource set changes.

Datasets are given diachronic identifiers, and linked to their temporal versions. Low-level and high-level changes are computed between versions. We propose a set of rules to map such models to the 2x2 space model. For relational models, we extend

R2RML[1], where relations are mapped to classes and columns to properties, as schema objects. Tuples create records and fields create record attributes. Multidimensional models are modeled as profiles of the Data Cube Vocabulary. Measures, dimensions and attributes are mapped to properties and observations create records and record attributes. In ontologies, classes and properties are typed as schema objects while triples create record attributes. Groups of triples with the same subject create records.

5 Conclusions

In this paper, we have presented our position towards evolution management on the Data Web, as well as the challenges and requirements for preservation and evolution management of heterogeneous web datasets. We have proposed a model for evolution, the components of which can reside into a 2X2 space where objects are separated by their temporal dependence and their curator-imposed evolution semantics and showed how to map three common models to this.

6 References

1. James, M. et al. "Big data: The next frontier for innovation, competition, and productivity." (2011).
2. Stavrakas, Y. et al. "Diachronic Linked Data: Towards Long-Term Preservation of Structured Interrelated Information." *arXiv preprint arXiv:1205.2292* (2012).
3. Van de Sompel, H. et al. "An HTTP-based versioning mechanism for linked data." *arXiv preprint arXiv:1003.3661* (2010).
4. Van de Sompel, H. et al. "Memento: Time travel for the web." *arXiv preprint arXiv:0911.1112* (2009).
5. Dyreson, C. et al. "Managing versions of web documents in a transaction-time web server." In *Proceedings of WWW2004*, pp. 422-432. ACM (2004).
6. Raymond W. and Lam, N. "Managing and querying multi-version XML data with update logging." In *Proceedings of the 2002 ACM symposium on Document engineering*, pp. 74-81. ACM (2002).
7. Buneman, P. et al. "Archiving scientific data." *ACM Transactions on Database Systems (TODS)* 29, no. 1 (2004).
8. Papastefanatos, G. et al. "Capturing the history and change structure of evolving data." In *Proceedings of DBKDA 2013*, pp. 235-241 (2013).
9. Umbrich, J. et al. "Towards dataset dynamics: Change frequency of linked open data sources." (2010).
10. Völkel, M., and Groza, T. "SemVersion: An RDF-based ontology versioning system." In *Proceedings of the IADIS international conference WWW/Internet*, vol. 2006, p. 44 (2006).
11. Popitsch, N., and Haslhofer, B. "DSNotify: handling broken links in the web of data." In *Proceedings of WWW2010*, pp. 761-770. ACM (2010).
12. Umbrich, J. et al. "Dataset dynamics compendium: A comparative study." (2010).
13. Papavasileiou, V. et al. High-level change detection in RDF(S) KBs. ACM Trans. Database Syst. 38(1): 1 (2013).

[1] http://www.w3.org/TR/r2rml/