Semantic Exploration of Archived Product Lifecycle Metadata under Schema and Instance Evolution

Jörg Brunsmann

Faculty of Mathematics and Computer Science, University of Hagen, D-58097 Hagen, Germany
joerg.brunsmann@fernuni-hagen.de

Abstract. The product lifecycle spans from idea generation, design, manufacturing and service to disposal. During all these phases, engineers use their tacit knowledge to fulfill their tasks. If engineers retire or leave a company, their embodied knowledge also resigns. To circumvent such loss of important company's intellectual property, the engineer's knowledge is captured as linked data and then used as annotation for product lifecycle data models. To enable the reuse of data not only in the near-term, the product data and its annotated metadata are ingested into special long-term archives. However, achieving full preservation of semantically enriched product data requires the consideration of the linked data lifecycle which includes the evolution of schemas and instances. Such conceptualization and terminology changes pose the threat of semantic obsolescence of archived product data. Therefore, this paper describes dedicated metadata preservation functionality which respects knowledge evolution of the linked data lifecycle.

Keywords: Product lifecycle management; Linked data; Long-term preservation; Metadata; Schema evolution.

1 Introduction

Products are designed, manufactured and operated with complex, collaborative and knowledge intensive processes using tools provided by product lifecycle management (PLM) systems. During all PLM phases various actors create a large amount of heterogeneous digital product data. Automatically and manually captured metadata expressed as RDF based linked data [1] is used to annotate product models. In order to provide meaning for metadata, it is described by domain schemas which themselves are expressed in the RDF schema language which provides a vocabulary indicating how elements are to be interpreted as classes and properties.

When a product line reaches its end of life, many manufactured physical products (e.g. airplanes) might still be in operation for several coming decades in which the availability and understandability of the associated product data and metadata has to be guaranteed [13]. Due to the following legal and business reasons, the annotated product data models have to be archived and preserved for later reuse in several product lifecycle phases by various actors:

- an innovation lab engineer reuses ideation metadata to search for similar ideas that were rejected or not realized
- a design engineer reuses collaborative design rationale metadata for a product variation in order to avoid design mistakes
- an engineer compares the fuel consumption of simulated engine runs and the actual fuel consumption to validate the simulation model parameters
- a newly employed engineer reuses previously conducted and archived social search knowledge
- an engineer reuses service experiences and knowledge which is expressed as metadata for product improvements [6]
- an engineer reuses metadata which was inferred from sensor data for process improvements
- an accident investigator exploits project and provenance metadata during accident examination for social network or project organization knowledge
- a service mechanic searches spare parts according to an archived product part specification (product catalogue) which is described by metadata

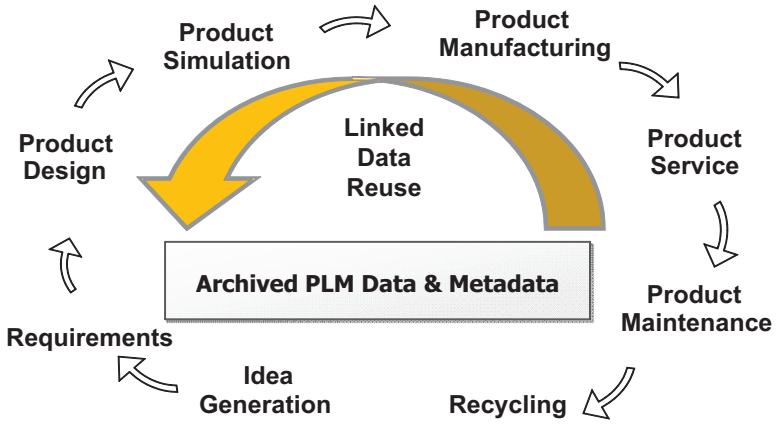

Fig. 1. The product lifecycle and linked data reuse.

Archived linked data is reused anticlockwise in the same or in a previous PLM phase (Figure 1). Linked data reuse is the last phase of the linked data lifecycle which is different from the product lifecycle. The linked data lifecycle spans from creation, annotation, archival up to the final goal of reuse. The reuse of linked data can be cumbersome, because vocabularies and linked data instances evolve due to changes in real-world phenomena. This knowledge evolution might lead to the loss of interpretation and traceability of archived data. Therefore special functionality is needed to preserve metadata under schema and instance evolution.

The remainder of the paper is structured as follows. The next section provides a characterization of the linked data lifecycle in the context of archival of semantically annotated product data models. Section 3 proposes a semantic digital archive system architecture that respects knowledge evolution and Section 4 describes an example scenario of domain schema evolution. The last section describes future work.

2 Linked Data Lifecycle

Linked data instances conform to vocabularies that make common domain knowledge explicit and usable for machines and humans. Vocabularies are expressed as schemas that enable interoperability of systems, actors, tools as well as interoperability with the future. Therefore linked data is suitable for expressing knowledge that is created during the lifecycle of a product. This section describes an idealized lifecycle of linked data in the context of semantic digital archives including the phases of capturing, annotation, archival, evolution, preservation, exploration and reuse.

2.1 Capturing and Annotation

The creation of linked data is done either automatically or manually. Automatic metadata extraction must be executed in real time because it cannot be recreated later on (e.g. simulation run with specific model parameters or metadata for project meetings). Manual metadata capturing has been implemented for the PLM environment ARAS Innovator [5] where a user is able to browse a schema and select a linked data instance as annotation of a product data model entity. This paper, does not consider the important automatic and manually metadata extraction in more detail.

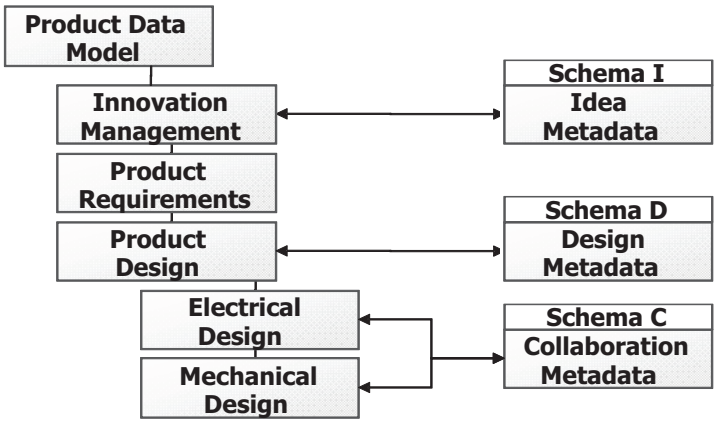

Fig. 2. An annotated product data model

Although linked data is data on its own, it can be regarded as metadata when it is used as annotation of other data. For example, product data model entities of different product lifecycle phases can be annotated with linked data (Figure 2). The product data model has entities (product part, 3D file, requirements document etc.) that describe the different PLM phases and these entities are annotated by linked data instances that conform to domain schemas. The annotated product data model is held in special repositories while the metadata can be stored and maintained external to the PLM repository. The metadata is referenced by using a unique URI and conforms to independently evolving domain schemas.

2.2 Archival

The time of archival of the product data model depends on the product lifecycle. When a product reaches its end of life, the product data model and its annotations are ingested into a long-term archive. Product lifecycle metadata is maintained by *External Systems* (ES) including collaboration capturing tools, MCAD and ECAD applications, design rationale capturing tools, etc. Figure 3 depicts the workflow execution in a PLM environment that includes long-term preservation functionality.

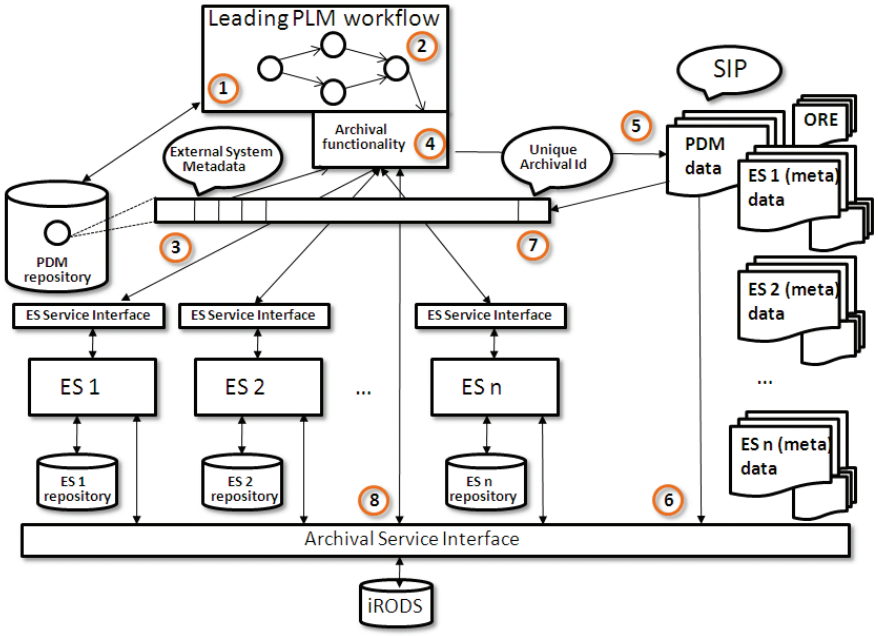

Fig. 3. Integration of long-term preservation functionality into PLM processes.

First of all, the PLM workflow (1) execution triggers the archival functionally (2) at special point in time (end of life, release for production). Because the extensible product data model contains references to external systems (3) that implement a special service interface, the archival functionality iterates over the connected external systems and collects all product relevant metadata (4). The collected data and metadata is then aggregated into an OAIS based SIP (Submission Information Package) [9] described by OAI-ORE based packaging information (5). The SIP processing also includes a normalization of data and metadata. The data normalization transfers a proprietary product data model into a standard product data model (e.g. PLCS or PLM/XML). The metadata which conforms to an external schema can be semantically normalized into metadata conforming to an archive local schema which makes preservation more controllable. In addition, metadata can also be syntactically normalized (e.g. N3, KIF). The whole product data collection is then ingested into a long-term archive (6). The long-term archive returns a unique id for the ingested

product data which is stored in the product data repository (7). The product data model might be deleted from the active repository. Finally, by using the long-term archive access interfaces, product data models can be queried and accessed (8).

2.3 Evolution

Linked data model real world domains which are continually changing especially in the engineering realm due to technology innovations and knowledge explosion. The data instances and their associated schemas must reflect these changes. New versions of existing schemas are generated or new domain schemas are being invented. Such semantic heterogeneity poses a threat for archived linked data and also archived queries may become invalid. Therefore, the software application EVO (Evolving ontologies) has been implemented that allows semi-automatic generation of schema and instance mapping when a new version of a schema is generated. In addition, the tool allows detection of mapping inconsistencies during editing schema updates and it allows capturing the rationale and the provenance of schema updates. Finally, the visualization (timeline widget) of schema elements updates and instances is possible. Since the mappings are stored in the same named graph as the schema and the instances are described by a dedicated vocabulary, they are operational and can be exploited during preservation.

2.4 Preservation

In the engineering domain, the preservation of CAD data [8] and the implementation of format registries [4] are of great importance. While these aspects are topics in other research projects, the preservation of metadata (e.g. product categorization ontologies [12]) is as important as the preservation of data but has not been considered in great detail [2]. Therefore during evolution of data and schemas, mappings are generated (see above) which can be used to preserve metadata. The preservation of metadata must include migration functionality as OAIS extension [10] which requires that an operational change set is identified during schema and instance evolution. The change set can be pulled or pushed upon request from administration. After retrieval, the change set can be stored or executed immediately. The change sets can also be used to migrate archived SPARQL queries. After migration, metadata and queries conform to a new version of the same metadata schema or to another domain vocabulary.

2.5 Exploration and Reuse

When an archived product data model is accessed, it is likely that the archive consumer does not have an idea what has been ingested. The consumer only knows the goal of his archive exploration ambition. By using domain schemas and data instances to annotate a product model, it can be easier understood by future archive consumers. When using an archive in this standalone fashion, archived schemas might be used for exploration. However, semantic archives can also be integrated in daily

business workflows (e.g integrated as an active repository into the PLM processes). Then, the integration of semantic digital archives faces the problem of evolving domain vocabularies. Fortunately, schema mappings that have been generated during the evolution phase can be exploited for *query mediation* during exploration and for *metadata transformation* during reuse in a contemporary environment.

Query mediation performs searches via other schema versions or on other domain schemas by rewriting incoming SPARQL queries without migrating metadata. Backward vertical query mediation finds archived instances that conform to a contemporary schema while forward vertical query mediation finds contemporary instances that conform to an archived schema. Horizontal query mediation finds instances based on equivalent classes and properties of other vocabularies.

Metadata transformation carries metadata which conforms to schema X into metadata conforming to schema Y upon request by the consumer during archive access. Both schema X and Y describe the same domain but with different conceptualizations modeled by different schema engineers.

2.6 Summary

The sections above described the different phases of the linked data lifecycle and their connection to long-term archival functionality (see also Figure 4).

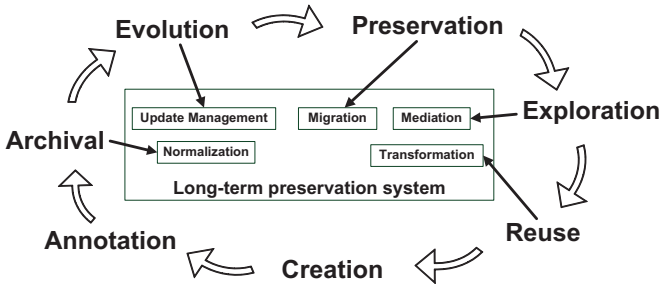

Fig. 4. The linked data lifecycle and long-term archival functionality.

During the pre-ingest phase, metadata is created and being used as annotation by a producer (e.g. engineer). At specific points in time, an engineer or administrator will syntactically and semantically normalize and archive the annotated product data models. A domain schema engineer is responsible for processing changes to data and schema. During this evolution, special tools collect operational change sets and push them to the archive or they are pulled by the archive. Upon request of administration, metadata is migrated within the archive or an archive consumer will explore the product data model by browsing the domain schemas and executing queries that might be mediated due to knowledge evolution. Finally, the archived metadata can be transformed during access of an archive consumer so that the metadata conforms to contemporary vocabulary. The transformation can be regarded as the creation of new metadata and the lifecycle starts from the beginning.

3 Semantic Digital Archive System Architecture

The previous section described the phases of the linked data lifecycle in the realm of the archival and preservation of product lifecycle data models. This chapter unites the functionality needed for handling the linked data lifecycle into a semantic digital archive system architecture that respects knowledge evolution. Figure 5 shows a system architecture of a semantic digital archive that is integrated into the daily business workflow. The architecture can be easily adopted for specific business workflows (e.g. PLM processes, library domain processes).

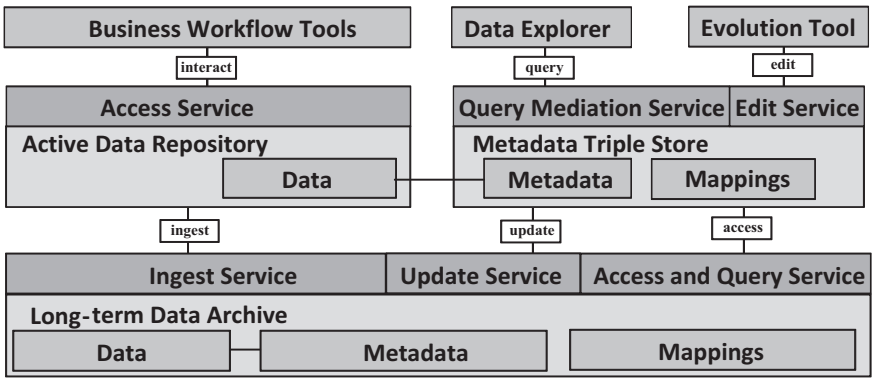

Fig. 5. Semantic digital archive system architecture respecting knowledge evolution.

The system architecture is made of three different layers:

Tool layer: the first layer contains the workflow tools (e.g. CAD design software) and a special data explorer tool that allows accessing the repository and the archive via browsing of metadata schemas. Finally, the evolution tool allows editing the metadata and the associated schemas. While editing the metadata, mappings are maintained semi-automatically.

Active (meta)data repository layer: the second layer contains the data repository and a triple store which holds the metadata. While the workflow tools interact with the data repository, the data explorer is able to query both the repository as well as the metadata repository because the metadata references the active data repository via annotations. By querying and finding metadata, product data model entities can be explored.

Archive layer: the third bottom layer contains the long-term archive functionality. The data from the active repository and the metadata is ingested into the long-term archive on demand when specific points in time of the business workflow are reached. The long-term archive also contains an access and query service that allows the data explorer to access the archived metadata. Finally, an update service is able to accept operational updates from the metadata triple store.

4 Example Usage Scenario

This section illustrates a knowledge evolution scenario (modeled as schema update) from the early ideation phase of a product lifecycle. Assuming, an engineer works for an innovation lab and he has to produce innovative and commercially attractive consumer electronic products. Innovation management software allows maintaining the idea semantics and visualization. Although nearly all of the ideas are not realized they are still important company intellectual property and therefore they are archived.

An idea contains among title, descriptive text, visual illustrations and creation date also the ideas' business category. The business category is a semantic annotation that includes concepts like Beauty Beverage Appliances, Shaving & Grooming, Kitchen Appliances, Sleep and Television. The following schema definition reflects the given scenario (namespaces prefixes are not shown).

RDFS definition excerpt of a product ideation vocabulary in the 1970s

```
:BusinessCategory a rdfs:Class .

:Television a :BusinessCategory .

:Idea a rdfs:Class .

:ThreeDIdeaFromThe70s a :Idea .

:hasBusinessCategory a rdf:Property ;
    rdfs:domain :Idea ;
    rdfs:range :BusinessCategory .

:ThreeDTVIdeaInThe70s :hasBusinessCategory :Television.
```

The schema defines the classes *BusinessCategory* and *Idea* and a property (*hasBusinessCategory*) that connects an idea with a business category. In addition, two instances are defined as *Television* (a business category) and a 3D TV related idea from the 1970s. Then, the schema and the instances are archived. Due to technology innovations, the business category *Television* has evolved into several categories (Figure 6), including the new class *ThreeDTV*.

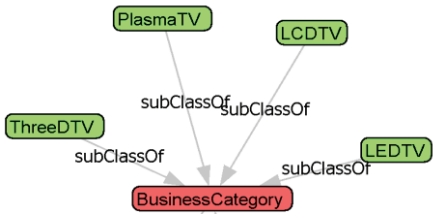

Fig. 6. Example schema evolution.

To prevent semantic obsolescence of archived ideas that contain the business category *Television*, the EVO tool allows defining a mapping between the newly introduced business categories and the previously defined category *Television*. Figure 7 shows the definition of the mapping between *ThreeDTV* and *Television*.

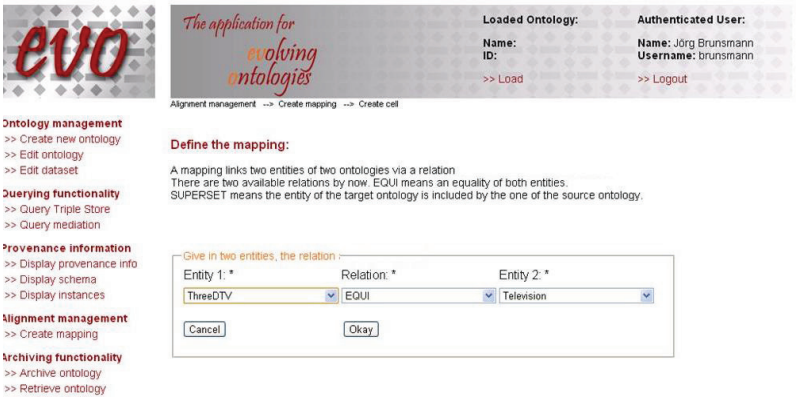

Fig. 7. Mapping definition between the 'ThreeDTV' and 'Television' business category.

Now, the engineer has a 3D TV related product innovation idea and he remembers that 3D TVs were already envisioned in the 1970s. The engineer wants to explore the active idea repository as well as the long-term archive in parallel because he don't want to reinvent the wheel or the same idea was probably already rejected for some reason or the engineer wants to get inspirations by studying similar ideas.

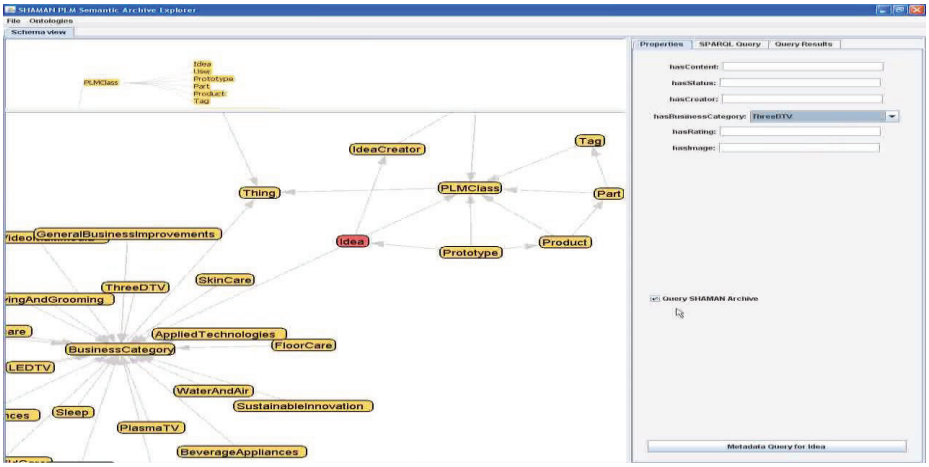

Fig. 8. Semantic exploration of archived product data under knowledge evolution.

The engineer uses a special semantic exploration tool, to search for ideas (Figure 8). This tool shows on the left-hand side a domain schema which has been loaded

from the archive or from the repository as a two dimensional graph. By selecting a class from the schema, its properties are displayed on the right-hand side. For example, the list of business categories can be selected from a drop down box. Since the schema has evolved, the *Television* business category is not available any more (only *ThreeDTV*). Fortunately, a checkbox can be used to indicate that the search should also be executed in the archive. By doing so, the previously defined mappings between the business categories can be exploited so that archived ideas conforming to *Television* category are also part of the result set.

5 Summary and Outlook

While [3] described a high level integration of long-term archival functionality into PLM processes, this paper derived a semantic digital archive system architecture respecting knowledge evolution by investigating the linked data lifecycle. First, metadata is created and used as annotation. Then, during archival, metadata is syntactically and semantically normalized before it is ingested. Upon request, the metadata and queries can be migrated within the archive. While searching for archived data, the incoming queries can be mediated without migrating the metadata. Finally, during reuse the metadata can be transformed to contemporary schemas. The migration, transformation and mediation functionality depend on operational change set that have been collected during linked data evolution.

Future work includes the evaluation of the linked data lifecycle not only for semantic archives but also for single web sites and the whole web of data. Also, the preservation functionality is currently being implemented as standalone prototype application. The integration as metadata services into an OAIS archive has to be done. In addition, a three dimensional interface for browsing and understanding archived schemas can be evaluated. Finally, annotated RDF [11], multidimensional RDF [7] or the approach described in [14] can be explored for archival of instance evolution.

Acknowledgments. This paper is supported by the European Union in the 7th Framework within the IP SHAMAN [3].

References

1. Bizer, C., Heath, T., Berners-Lee, T.: Linked Data - The Story So Far. International Journal on Semantic Web & Information Systems. Vol. 5, Issue 3, pp 1-22 (2009)
2. Brunsmann, J., Wilkes, W.: State-of-the-art of long-term archiving in product lifecycle management. International Journal on Digital Libraries, Special Issue on Persistent Archives (2011)
3. SHAMAN Project: SHAMAN Homepage. htp://shaman-ip.eu (2009)
4. KIM Project. Knowledge and Information Management Grand Challenge Project. http://www-edc.eng.cam.ac.uk/kim/(2009)
5. Aras Innovator Homepage: http://www.aras.com (2011)

6. Brunsmann, J., Wilkes, W., Brocks, H.: Exploiting Product and Service Lifecycle Data. 8th International Conference on Product Lifecycle Management, Eindhoven, Netherlands, July 11-13 (2011)
7. Gergatsoulis, M., Lilis, P.: Multidimensional RDF. In Proc. 2005 Intl. Conf. on Ontologies, Databases, and Semantics (ODBASE), Vol. 3761, Springer, 1188–1205 (2005)
8. LOTAR Project. Long-Term Archiving and Retrieval. http://www.lotar-international.org (2011)
9. CCSDS Reference model for an Open Archival Information System (OAIS). Blue Book, Consultative Committee for Space Data Systems. Also published as ISO 14721:2003. http://www.ccsds.org/documents/650x0b1.pdf (2002)
10. Brunsmann, J.: Product Lifecycle Metadata Harmonization with the Future in OAIS Archives. International Conference on Dublin Core and Metadata Applications, The Hague, Netherlands, September 21-23 (2011)
11. Lopes, N., Polleres, A., Straccia, U., Zimmermann, A.: AnQL: SPARQLing Up Annotated RDFS. In The Semantic Web - ISWC 20010, 9th International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11 (2010)
12. Hepp M., Leukel J. and Schmitz V.: A Quantitative Analysis of Product Categorization Standards: Content, Coverage, and Maintenance of eCl@ss, UNSPSC, eOTD and the RosettaNet Technical Dictionary (2007)
13. Heutelbeck, D., Brunsmann, J., Wilkes, W. Hundsdörfer, A.: Motivations and Challenges for Digital Preservation in Design and Engineering. First International Workshop on Innovation in Digital Preservation, Austin, Texas, USA, June 19 (2009)
14. McBride, B. Butle, M.: Representing and Querying Historical Information in RDF with Application to E-Discovery. 8th International Semantic Web Conference, Washington, USA, October 25-29 (2009)