Enabling Customization of Document-Centric Systems Using Document Management Ontology

R. Molnar*, S. Gostojić*, G. Sladić*, G. Savić*,Z. Konjović*
*University of Novi Sad/Faculty of Technical Sciences, Novi Sad, Serbia
{rmolnar, gostojic, sladicg, savicg, ftn_zora}@uns.ac.rs

Abstract - This paper introduces a conceptualization of the document management domain that is serving as a foundation for a semantically-driven document management system. The conceptualization is based on the ISO 82045 family of standards for document management and is specified in OWL. Legislative documents were used as a case study and a proof of concept of the proposed conceptualization.

I. INTRODUCTION

As the usage of Document management systems (DMS) [1,2] increases throughout different sectors of the economy, the need for a cost-effective yet domain-specific DMS arises. DMS should enable simple capture, storage, transfer, retrieval and browsing of documents and provide services such as metadata capture, security, integration and version control [3,4]. However, most of the current DMS implementations do not offer domain-specific services (they lack domain semantics) since they are difficult to customize to a particular domain.

Introduction of semantic technologies [5] into document management systems can mitigate those shortcomings by providing an abstract domain-independent model which can be easily adapted to a concrete domain and serve as a foundation for the semantically driven document and workflow management system described in [6].

The semantic model (i.e. ontology) identifies common features of documents belonging to different domains. The fact that domains and documents belonging to those domains differ in their characteristics does not affect the complexity of the proposed model because only the common features, such as document types, document structure, metadata, classes, and identifiers, were modeled. Most of the concepts are co-opted from the ISO 82045 family of standards [2] which is de facto and de jure standard in the document management domain. The model is specified in OWL DL dialect [7] in order to provide the maximum expressiveness possible while retaining computational completeness. The flexibility of the abstract model allows extension with concrete domain-specific elements. Those elements enable implementation of domain-specific services offering additional features. At the same time, the existing features are not compromised, and there are no new constraints on generic documents. A piece of legislation (i.e. a law) was used as a case study of the flexibility of the proposed model. Some important domain-specific services in the legislation domain, which are based on the concrete domain-specific elements, are judgement anonymization and retrieval and browsing of legislation.

II. RELATED WORK

MAchine Readable Cataloging (MARC) [8] is a bibliographic standard that applies to document management by representing the document metadata in a machine-readable form. It is primarily designed to enable exchange of documents between the systems. This standard is an outdated version of the archaic card catalogs. Today's information needs are more demanding than before as there is more than one type of media to be described and managed.

Digital Object Identifier (DOI) [9] framework is used to identify digital objects. It features persistence, network accessibility and interoperability with other systems. It is used by other systems that provide domain-specific identifiers, such as CrossRef [10] and DataCite [11]. The DOI system was initiated in 1998 and has later been standardized as ISO 26324. The DOI can be used to identify any digital object, but the major disadvantage of this system is difficult registration of a DOI name. Unique identification of a digital object is done by Registration Agency and it is not free.

Metadata Encoding and Transmission Standard (METS) [12] is a specialization of MARC standard. METS uses Uniform Resource Identifiers (URI) to identify components that, along with relationships between them, can form one digital entity. Unfortunately, the flexibility of the standards can cause many problems with interoperability as the very same digital object can be represented in many different ways. Such approach makes difficulties in basic information retrieval operations like indexing and searching.

MARC, DOI and METS mostly deal with management of metadata.

MoReq2010 [13] is a comprehensive specification of functions, services and processes that records management systems should support. The specification does not specify which algorithms should be used, but it requires compliance to various registry formats. Records management systems focus on ensuring authenticity, integrity, usability, and reliability to make sure that the records are always available and kept as long as it is necessary. This specification can be used as a very good guideline for the creation of a well-formed electronic document management system. Since mid-2011, when the specification has been published, only one software product has passed a compliance testing and is MoReq-certified.

Ontalk [14] is a document management and retrieval tool that includes semi-automatic metadata generator and an ontology-based search engine. It is based on three on-

tologies: the document schema, the document type, and user domain ontology. The system does not cover document classification and is platform-dependent due to the technologies used to implement it. Also, the search engine works only with properly annotated documents that have to be annotated by the user.

ISO 82045 family of standards defines document management concepts and establishes document management principles covering all the phases starting from the conceptual idea of the document to its deletion. Orthogonal features, such as version control and security, are also in the scope of the standard. Part 1 of the standard is intended to be supported by computer-based systems such as DMS or Product Data Management Systems (PDMS).

III. Document Management Ontology

The reviewed document models are not flexible enough to allow multiple extensions with domain-specific rules. Furthermore, most of them do not introduce the concept of versioning and lack the support for document life-cycle management without which it is not possible to achieve management of documents.

The proposed ontology is strongly influenced by the ISO 82045 family of standards and imports time and provenance related concepts from Time Ontology [15] and PROV-O [16] respectively. It is designed to support standard DMS function and enable easy integration with business processes implementing document life-cycle.

ISO 82045 family of standards defines concepts such as the document, the part of a document, document metadata, the document version, the document relationship, the identification of the document, and the classification of a document. Those concepts and their relationships are shown in Figure 1. It is possible to choose desired level of detail by using only necessary concepts (i.e. neglecting the concepts that are not of interest to the user at the given time).

Each document is an instance of exactly one document type that can be inferred from the relationships it has with metadata and other documents. Document types, defined in plain text and OWL as fragments of expressions written in Manchester syntax [17], are as follows:

SingleDocument (Listing 1) –An identified object associated with metadata whose content is entirely contained within the object (e.g. a note, a picture).

```
Document and
hasFragment some DocumentFragment and
hasMetadata some Metadata and
hasPart exactly 0 Thing
```

Listing 1. Single document

CompoundDocument (Listing 2) –An identified object associated with metadata that has content and contains another document without associated metadata (e.g. a report showing a table or a graph). As defined in [2], a compound document is a document containing documents (parts) that cannot be separately identified and cannot be independently managed.

```
Document and
hasFragment some DocumentFragment and
hasMetadata some Metadata and
isComposedOf (Document and hasMetadata
exactly 0 Metadata)
```

Listing 2. Compound document

AggregatedDocument (Listing 3) – An identified object associated with metadata that may have content and contains other documents with associated metadata (e.g. the web site). In other words, an aggregated document is a document containing separately identified documents (parts) that are logically dependent but can be physically independently managed.

```
Document and
hasFragment onlyDocumentFragment and
hasMetadata some Metadata and
isAggregatedOf (Document and hasMetadata
some Metadata)
```

Listing 3. Aggregated document

DocumentSet (Listing 4) – An identified object with associated metadata that has no content (all content is contained in other documents that are part of the document set). The relationship between the documents is implemented in the same manner as in the *AggregatedDocument*. As defined in [2], a document set is a collection of documents that are managed together as a unit for a specific purpose.

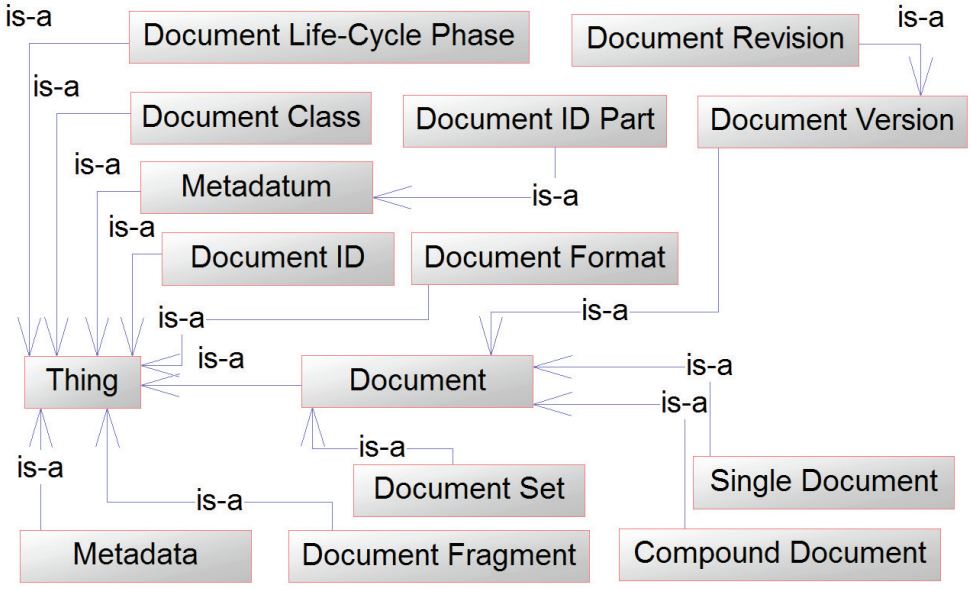

Figure 1.Document management concepts and their relationships

```
Document and
hasFragment exactly 0 Thing and
hasMetadata some Metadata and
isAggregatedOf (Document and hasMetadata
some Metadata)
```

Listing 4. Document set

Properties isAggregatedOf and isComposedOf are subproperties of the hasPart property. Composition relates compound document to their parts and is used when parts are not identified and managed independently. Aggregation relates aggregated documents to their parts and is used to aggregate individual documents into a group which can be described with an additional content and metadata (similarly to a document set that cannot have its own content).

DMS should also support metadata management. Although documents do not necessarily have associated metadata, the proposed ontology supports attaching metadata belonging to various schemata to documents. Since metadata management is out of the scope of this paper, the part of the ontology that deals with metadata is simplified as much as possible. *Metadata* class is implemented as a collection of individual metadata entries. Those entries are individuals of *Metadatum* class that represents a key-value pair.

Each document should be uniquely identified, regardless of whether it is created by the system or another system in its environment. Identification mechanism can be simple or complex, depending on the needs of the user or external identification mechanisms. Each identifier is composed of one or more parts that are individuals of *DocumentIDPart* class, a subclass of *Metadatum* class. Dublin Core Metadata Initiative (DCMI) [18] is just one of many standards proposing that identifiers should be viewed as metadata.

Document classification is another essential feature of document management. Since documents may belong to one or more classes simultaneously (*DocumentClass* class), there is a need to rely on external classification systems. One of the many possible systems is described in [19].

Document content(or document version content) can be either unstructured or structured. In the first case, the content is contained within the document itself. In the other case, the content is contained within document fragments (the individuals of *DocumentFragment* class). The document is structured by defining a hierarchy (isFragmentOf or hasFragment properties) and order (isAfter or isBefore properties) among the fragments. Although those properties should be transitive, due to the computational complexity of the resulting ontology (it would be out of the scope of OWL DL dialect) they are not implemented as such.

The*DocumetnFormat* class is used to specify document format. A document or a document version can be represented in multiple formats (e.g. Microsoft Office Word, PDF, and HTML).

The Time Ontology is an ontology of temporal concepts. The ontology was created as a mean of unifying time-related data that can be found on the internet. The main class of this ontology is the *TemporalEntity* that has two subclasses: the *Instant* and the *Interval*. The *Instant* represents an exact moment in time while the *Interval* represents a time interval. Each interval is determined with its beginning and its end (it can be described as two individuals of the class *Instant*). The interval can also be positively or negatively infinite. A negatively infinite interval is an interval without the beginning, and a positively infinite interval is an interval without an end.

The PROV-O is another World Wide Web Consortium (W3C) recommendation that enhances functionality and interoperability of systems by introducing classes, properties and restrictions used to represent and exchange provenance information. Provenance is defined as information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness [20]. Three main classes of the model are *Agent*, *Activity,* and *Entity.* Their relationships are shown in Figure 2.

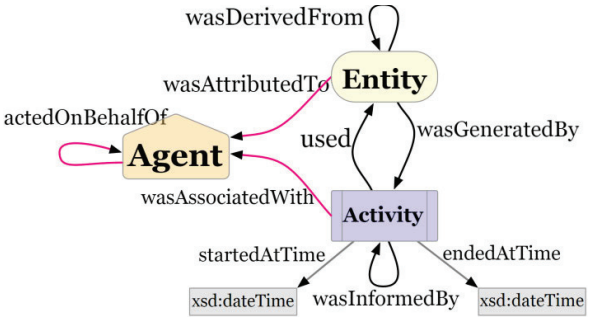

Figure 2. Main PROV-O concepts [16]

The Time Ontology and the PROV-O ontology are used because they are W3C recommendations and have been proven in practice.

Depending on the definition, a new version of the document does not necessarily imply that its content has been changed. It might imply that its presentation did. Since the proposed ontology does not cover the document presentation layer, we opted for an approach in which two document versions differ only by the data they contain. In order to enable versioning of documents, the *DocumentVersion* class is defined as the subclass of the *Entity* class (an entity is a physical, digital, conceptual, or another kind of thing with some fixed aspects). Since document versions also pass trough most of the life-cycle phases as documents themselves, the *DocumentVersion* class is also a subclass of the *Document* class.

In the case of sequentially effective versions, the latest released document version is the only operative, and it serves all intended purposes of all previous document versions. Nevertheless, a document may have more than one effective version at a time (Figure 3). Concurrently effective versions assume that multiple versions of the document are operative, at a particular moment in time, and each effective version still serves its defined purpose [2].

DocumentRevision, which is a subclass of *DocumentVersion* class, is defined as an officially confirmed document version.

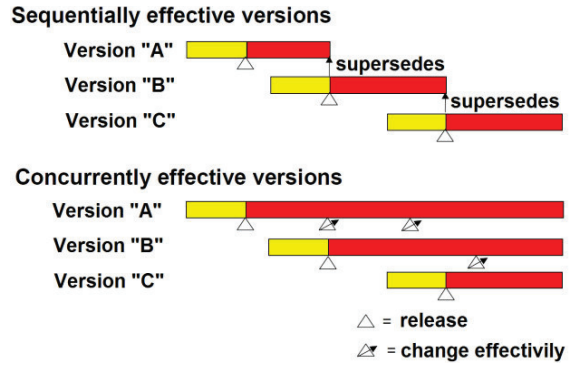

Figure 3. Version effectivity [2]

Document life-cycle has multiple phases. Each of the phases represents a state of the document in time. There are seven phases: Initiation, Preparation, Establishment, Use, Revision, Withdrawal, and Deletion [2] (Figure 4).

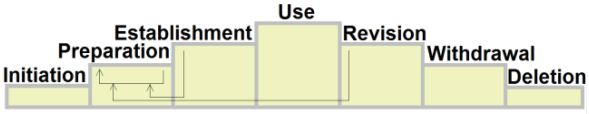

Figure 4.Document life-cycle phases [2]

In the initiation phase, the document is uniquely identified and classified within the system. DMS may use its subsystems or external systems in order to properly identify and classify the document.

The preparation phase is reserved for document content development after which it enters an establishment phase where the document undergoes various checks and approval within the responsible organization. When a document enters the approval process, all changes are traceable, and the document should already be under version control.

The use phase starts as soon as the document is completely verified and released.

During the revision phase, the document content is being changed.

After the document becomes useless, it is then withdrawn. The document itself will be kept for a minimum legally required period (that may vary) as an archive.

Deletion phase is the last phase in document life-cycle, and it means that the document is being completely deleted, and it can no longer be traced. In some specific cases, complete elimination is not possible as there are active references to the document. In such cases, the document metadata is kept only in order to keep the references correct [2].

IV. Case Study

Legal profession is a sector of the economy that uses huge quantity of documents. Apart from having complex structure and being interdependent, those documents are characterized by strict identification mechanisms, metadata and classification schemata, and changing life-cycle management processes. Therefore, we decided to use legislation (statutes, laws) as a case study and proof of concept of the extension and instantiation of the proposed ontology. The abstract document management ontology and its instantiation in the legislative domain are available at [21].

As an example, we instantiated "Law on Personal Data Protection" [22] in two versions (as enacted in 2008 and 2009) in several different formats. Since the legislation is highly structured, the content of those versions is associated with *DocumentFragment* instances that are ordered in the proper manner.

Each law passes through several life-cycle phases before it is used.

In the initiation phase, the bill gets a unique identifier and one or more classifications within some classification system.

There are several mechanisms to identify the bill. The mechanism used in Serbian legislature relies on its name and the volume and the number of the official journal in which it is published. One legal document identification mechanism is Uniform Resource Name Namespace (URN) for Sources of Law (LEX) [23]. The identifier is conceived in such manner to depend only on the document characteristics and is independent of the document availability, access mode and physical location [23]. It has to be: globally unique, transparent, persistent, location-independent and language-neutral. Those characteristics provide mechanisms of stable cross-country references. Another legal document identification mechanism is specified in Architecture for Knowledge-Oriented Management of African Normative Texts using Open Standards and Ontologies (AKOMA NTOSO) [24]. The syntax of AKOMA NTOSO identifier is based on Uniform Resource Location (URL) standard [25]. With both mechanisms, it is possible to differentiate the identifiers of the level of FRBR work, expression, manifestation and item [26].

A legal act can be classified according to its subject matter. In Serbian legislature, there are eighteen classes of a legal act according its subject matter. Some of them are: defense, military and internal affairs; justice, criminal law and proceedings; trade, tourism and hospitality; public institutions, science, education, culture, media and sport; and many other. All of these classes have their subclasses. For example, trade, tourism and hospitality has four subclasses: trade, procurement and consumer protection; tourism and hospitality; protection of competition and state aid control; and stockpiles. The number of subclasses goes upto 23 [27].

After initiation phase, the document enters the preparation phase, i.e. the bill is drafted. As the content of the bill is strictly structured, it is possible to distinguish more than a few elements of the structure hierarchy: part, chapter, section, subsection, article, item, point, subpoint and line [25]. In our case, only articles and items appear as individuals of *DocumentFragment* class. It is important to notice that the model does not indicate whether the fragment is an article or an item. That can be infered from the name of the individual or by its place in the hierarchy.

The establishment phase it the phase in which the document, in this case, the bill, get enacted into law by the parliament, promulgated by the president, and published in the official gazette.

The use phase of a law in Serbia begins eight days after the law has been published at which point the law has legal consequences.

Document maintenance is carried out in the revision phase by enacting changes to existing legislation. As an example, we presented two versions of the law. These

versions are sequentially effective. In most cases, different versions of laws are sequentially effective, but there are some situations when there are two concurrently effective version of a law.

The law is repealed in the withdrawal phase.

V. CONCLUSION

In this paper, we presented some of the problems faced during design and development of DMSs and proposed a solution that is based on semantic web technologies. Documents were modeled starting from the concepts defined in the ISO 82045 family of the standards and time and provenance related concepts imported from Time Ontology and PROV-O.

Nevertheless, some problems still remain to be solved. There is a need to merge the presented ontology with metadata and classification ontologies. Furthermore, the static conceptualization of the document management domain has to be rethought in the context of business processes and workflow management. Enriching the model with Friend of a Friend (FOAF) is considered. FOAF is well-known ontology describing people and their relationships which can be used to improve security data. Also, our main goal, i.e. to customize the model to domains other than law and to implement components of the DMS described in [6], still has to be achived.

REFERENCES

[1] F. Castillo-Barrera, H. Durán-Limón, C. Médina-Ramírez, B. Rodriguez-Rocha, "*A method for building ontology-based electronic document management system for quality standards - the case of the ISO/TS 16949:2002 automotive standard*", Applied Intelligence, vol. 38, pp. 99-113, 2013

[2] International Organization for Standardization (ISO), "*ISO IEC 82045-1: Document Management – Part 1: Principles and Methods*," ISO, Geneva, Switzerland, 2001

[3] H. Zantout, F. Marir, "*Document Management Systems from current capabilities towards intelligent information retrieval: an overview*", International Journal of Information Management, vol. 19, Issue 6, pp. 471-484, 1999.

[4] A. Azad, "*Implementing Electronic Document and Record Management Systems*", chapter 14, Auerbach Publications, ISBN-10: 084938059, 2007

[5] T. Berners-Lee, J. Hendler and O. Lassila, "*The Semantic Web*", Scientific American, 2008.

[6] S. Gostojic, G. Sladic, B. Milosavljevic, M. Zaric and Z. Konjovic, "*Semantic Driven Document and Workflow Management*", International Conference on Applied Internet and Information Technologies (AIIT), 2014

[7] OWL DL [online], Avaliable at: http://www.w3.org/TR/2004/REC-owl-semantics-20040210/rdfs.html#5.4 [accessed January 14, 2015]

[8] MAchine Readable Cataloging (MARC) [online], Available at: http://www.loc.gov/marc/ [accessed January 14, 2015]

[9] Digital Object Identifier (DOI) [online], Available at: http://www.doi.org/ [accessed January 14, 2015]

[10] CrossRef [online], Available at: http://www.crossref.org/ [Accessed 20 Dec. 2014]

[11] DataCite [online], Available at: https://www.datacite.org/about-datacite/what-do-we-do [accessed January 14, 2015]

[12] R. Guenther, S. McCallum, "*New Metadata Standards for Digital Resources: MODS and METS*", Bulletin of the American Society for Information Science and Technology, pp12-15, ISSN: 0095-4403, 2003

[13] The DLM Foundation, "MoReq2010: Modular Requirements for Record Systems - Volume 1: Core Services & Plug-in Modules", [online], Available at: http://moreq2010.eu/ [accessed January 14, 2015]

[14] H.L. Kim, H.G. Kim, K.M. Park, "Ontalk: ontology-based personal document management system", WWW Alt. '04, May 2004

[15] Time ontology [online], Available at: http://www.w3.org/TR/owl-time/ [accessed January 14, 2015]

[16] PROV-O ontology [online], Available at: http://www.w3.org/TR/prov-o/ [accessed January 14, 2015]

[17] OWL 2 Web Ontology Language Manchester Syntax (Second Edition) [online], Available at: http://www.w3.org/TR/owl2-manchester-syntax/ [accessed January 14, 2015]

[18] Dublin Core Metadata Initiative (DCMI) [online], Available at: http://dublincore.org/specifications/[accessed January 14, 2015]

[19] C. McGregor, O. Alonso, S. Alpha, S. Buxton et al., "*Oracle Text Application Developer's Guide, 10g, Release 1 (10.1)*", chapter 6 "*Document Classification*"

[20] PROV-Overview [online], Available at: http://www.w3.org/TR/2013/NOTE-prov-overview-20130430/[accessed January 14, 2015]

[21] Document Management Ontology [online], Available at: http://www.informatika.ftn.uns.ac.rs/82045-1 [accessed January 14, 2015]

[22] Narodna Skupština Republike Srbije, "*Law on Personal Data Protection*", Službeni glasnik Republike Srbije no. 104/2009

[23] P. Spinosa, E. Francesconi, C. Lupo, "*A uniform resource name (URN) namespace for sources of law (LEX)*", Internet Engineering Task Force, Fremont, 2011, available at: http://tools.ietf.org/html/draft-spinosa-urn-lex-04 [Accessed February 20, 2015]

[24] Akoma Ntoso [online], Available at: http://www.akomantoso.org/, [accessed February 20, 2015]

[25] S. Gostojic, "*Kreiranje i korišćenje digitalnih dokumenata pravne regulative*", Doctoral dissertation, University of Novi Sad, 2012

[26] International Federation of Library Associations and Institutions, "*Functional Requirements for Bibliographic Records*", International Federation of Library Associations and Institutions, The Hague, 2007, available at: http://www.ifla.org/en/publications/functionalrequirements-for-bibliographic-records [Accessed February 20, 2015]

[27] Narodna Skupština Republike Srbije, "*Constitution of the Republic of Serbia*", Službeni glasnik Republike Srbije no. 98/2006