A Meta-metadata Ontology Based on ebRIM Specification

Igor Cverdelj-Fogaraši, Goran Sladić, Stevan Gostojić, Milan Segedinac, Branko Milosavljević
{igor.fogarasi, sladicg, gostojic, milansegedinac, mbranko}@uns.ac.rs
Faculty of Technical Sciences, University of Novi Sad

***Abstract* — This paper describes an approach for enabling unified, document type independent semantic web-based reasoning over various metadata sources in a document management system. A comprehensive, yet concise non-domain-specific metadata ontology influenced by ebXML RegRep standard is proposed to serve as a semantic basis for other domain-specific metadata ontologies to be mapped. To overcome disparities between non-domain-specific and domain-specific metadata ontologies, SWRL rules are combined with OWL knowledge base, thus enabling comprehensive semantic web-based reasoning using an extended set of OWL axioms. Dublin Core metadata ontology is used as a case study of this research.**

I. INTRODUCTION

One major limitation of existing DMSs (Document Management Systems) is the lack of domain specific services (such as domain specific browsing and retrieval of documents, life-cycle management, etc.) leading to a complicated customization of DMS for a specific domain. Document management, per se, does not necessarily have to make use of semantic web technologies [1]. Semantic web technologies, on the other hand, can be used as powerful means for implementing semantically supported DMS. To achieve a complete semantic document management, a couple of very important things such as: document management (an abstract metadata and document model), document life-cycle management, business process management, etc. have to be implemented on a semantic level. In this paper we limit ourselves exclusively to metadata, which represents only one of the mentioned aspects of semantic DMS, while the other aspects of semantically supported document management will be the subject of our further work. Correcting the aforementioned DMS weakness is our main guideline in carrying out this research.

Looking at records stored within a DMS, various domain documents can often be found. Therefore, different metadata standards, and metadata element sets consequently, are generally used to identify document content. Sometimes even documents from the same domain, issued by different agencies, are identified with different metadata element sets. Additionally, more often than not, attributes from different metadata element sets might actually be referring to the very same thing. All these documents together with their matching metadata represent a large heap of not too useful information, in terms of semantic reasoning.

Our solution for DMS deficiencies is to introduce semantics into DMS by modeling documents, business processes and metadata using semantic web technologies [1]. This model should consist of two layers: an abstract layer which models abstract documents, business processes and metadata and a concrete, non-abstract layer which models domain specific documents, business processes and metadata. Using these models enables domain specific customization of DMS, providing a wide range of domain specific services [2].

In this paper we focus on metadata model for describing documents stored in a DMS. In order for our DMS to leverage all of the available information to the maximum extent, we implemented a non-domain-specific metadata OWL ontology [3], based on ebRIM (ebXML Registry Information Model) specification. This ontology should serve as a semantic basis for the other metadata ontologies. More specifically, this non-domain-specific ontology can be considered meta-metadata ontology for other domain-specific metadata ontologies to be mapped. The ontology we implemented represents a core ontology which allows unified, document type independent reasoning over heterogeneous metadata sources, thereby enabling the employment of implicitly contained, passive knowledge.

Although we decided to go for a more flexible semantic approach, the fundamental concepts of our solution are still largely based on ebRIM specification. There are several reasons for restricting our choice to ebRIM. Probably, the most relevant one is its ubiquitousness. Nowadays, there are many different domain-specific implementations of ebRIM specification widely available, which proved it to be applicable in various domains. Just to name a few: Geospatial information systems [4], Sensor web [5], Healthcare informatics [6], Document management systems [7], and the list goes on. Furthermore, many governmental bodies and industries are also prominent in ebXML (Electronic Business using eXtensible Markup Language) Registry adoption for electronic information management and dissemination [8].

Besides the above mentioned, another fact supposed to be in favor of ebXML Registry Information Model is that ebRIM specification is a major part of the ebXML RegRep [9] standard approved by OASIS; organization considered to be one of the largest standards consortiums for electronic commerce on the Web.

II. RELATED WORK

In this section we outline the background of the research, focusing mainly on studies using either ebXML RegRep or familiar ebXML standard, for solving related problems in different domains of application.

In paper [10], authors propose a solution for mapping CPP (Collaboration Protocol Profile) and CPA (Collaboration Protocol Agreement) specifications to OWL ontology with the aim of enhancing the semantics of ebXML in E-commerce domain. The effort has been made in order to facilitate discovery and communication between business partners, without having to exchange large amounts of data. CPP and CPA specifications are a major part of ebXML CPPA (Collaborative Partner Profile Agreement) standard, commonly used in E-commerce. ebXML CPPA is another OASIS standard, beside ebXML RegRep, which will be discussed in more detail.

The authors of another paper [11], propose a solution for semantic enhancement of ebRIM in order to support a development of SCRR (Software Component Registry and Repository) system built upon the ebXML infrastructure, and according to the ebXML RegRep standard. Methodology discussed in this paper is directed towards extending the semantics of ebXML Registry Information Model by introducing a software component attribute ontology as a means for providing semantically richer description of software components within the registry.

The solution proposed in [12] is based on a pragmatic, metadata oriented approach. In this paper, authors introduce document type ontologies to facilitate modeling of structured metadata definitions within DMSs. The solution presented in this paper is based on ebXML RegRep standard. In order to enhance DMSs and enable ontology based document management, the authors of the paper suggested mapping OWL constructs to ebRIM, conforming to OWL extension profile recommendations. According to the proposed solution, an ontology based classification of documents within a DMS is demonstrated in the paper.

Methodologies described in [4, 5] are oriented towards web service discovery as well as collaboration and interaction in context of B2B negotiations. Authors of both papers agree that WSDL description of web services does not necessarily provide enough semantics for a successful B2B marketing. Oftentimes, a high-level description of service instances is also needed. Both proposed solutions [4, 5] are also based on ebRIM. In paper [13], the authors suggested a strategy for improvement of web services discovery by introducing semantics in ebXML registries. They suggested storing semantics, in a form of OWL constructs within ebXML registries. This makes the businesses more easily recognizable, thus making the collaboration between businesses more efficient. Another paper [14] related to web service discovery, also addresses the ebXML enrichment issue. In this paper authors describe how registries can be enriched in order to describe web service semantics through OWL ontologies. They propose matching OWL constructs to ebXML classification hierarchies, and describe how expressed semantics can be queried through standardized queries, which are an integral part of ebXML facility.

Another approach of semantic tagging and discovery of services, as well as other resources is covered in lecture [15]. Problems discussed in the lecture involve structure mapping: querying across heterogeneous data structures, and conceptual mapping: using knowledge from domain ontologies in order to improve the results. The authors recommended using domain ontologies for metadata enrichment. This is accomplished by tagging metadata with concepts from widely accepted domain ontologies, which allows any new domain knowledge to be leveraged instantly. The work presented in this lecture is based on OWL extension profile of ebXML RegRep standard, mentioned earlier in this section.

The work reported in [16] is headed towards identifying the potentials for ebRIM usage in healthcare informatics. The main focus of the research is achieving a semantic interoperability amongst different healthcare systems. The authors of the paper describe how ebXML registry semantic constructs can be utilized for annotation, storage, retrieval and discovery of medical archetypes. The solution they propose includes the implementation of archetype metadata ontology. They also describe a strategy for accessing archetype semantics through standardized queries and ebXML query facility. Moreover, the authors suggest how archetype data can be efficiently retrieved from medical information systems by using ebXML messaging system, a standard way of exchanging messages between organizations.

III. EBRIM SPECIFICATION

ebXML registry plays a fundamental role in the ebXML architecture. It serves as an application gateway for a repository to the outside world, governing how parties interact with the repository. It provides a means for sharing of relevant company information in a form of business semantics, to relevant parties in a highly controlled manner. The ebXML registry can be considered an interface, independent of the underlying network protocol stack, for accessing and discovering shared business semantics [17]. The main purpose of ebXML registry is enabling business process integration between the interested parties.

ebXML RegRep is an open specification approved as an OASIS standard [18] for metadata and content management software. It is capable of managing diverse content such as documents, images, services, devices, assets, schemas, WSDL, ontologies, records [19]; which, among other aforementioned things is the reason why ebXML is relevant, and thus so important for document management systems. On the other hand, ebRIM specification is a substantial part of ebXML RegRep standard. As such, it provides a high-level blueprint for metadata in the ebXML registry [20]. Its elements do not represent repository content, but the content metadata. They provide a definition of metadata type and their relationships, at a higher conceptual level.

There are two integral parts of ebXML RegRep standard. The first one is a registry, and the other one is a repository. The repository stores digital content, while the registry stores corresponding metadata. Accordingly, there are two major types of resources: Repository Item, represents an object stored in a repository, and Registry Object, a metadata used by a registry to classify and manage repository items. The ebXML Registry Information Model defines classes and their relationships used for Registry Object metadata representation.

Since our approach is metadata oriented, in this paper we will focus on Registry Object as a key resource representing metadata. Repository Item will be introduced later in the next section in order to establish a connection between a document (Repository Item) and its metadata (Registry Object) in ebRIM ontology.

The ebRIM model is composed of several sub-models: Core, Association, Classification, Provenance, Service, Event and Cooperating Registries information model.

Although each one of them plays an important role in a description of ebXML registry; Core, Association, Classification and Provenance information models are of a special importance in a definition of ebRIM metadata model. Therefore, they will be described and taken into consideration in more detail in the next section which deals with the implementation of its ontological counterpart.

Beside RegistryObject class, being crucial for modeling metadata, some of the other, not less significant classes include: *ExternalIdentifier* class enabling advanced methods for registry object identification; *ExternalLink* class providing a mechanism for linking registry objects to arbitrary content; *ClassificationScheme*, *ClassificationNode* and *Classification* classes introducing domain-specific taxonomies between registry objects; *RegistryPackage* class allowing aggregation of logically related registry objects; *Slot* class providing an unrestricted extensibility of registry object attributes.

In addition to the above mentioned classes, there are several equally important classes such as: *User* (super class: *Person*), *Organization*, *EmailAddress*, *TelephoneNumber*, *PostalAddress* describing parties responsible for creating, publishing or RegistryObject maintenance. As well as other Core, Association, Classification and Provenance member classes not mentioned above.

IV. EBRIM ONTOLOGY

ebXML RegRep specification is reported to be designed for extensibility. However, although there is a couple of very useful extension profile specifications; neither one of them suggests a semantic approach, except for the ebXML RegRep OWL Profile. What we found as problem with OWL extension profile is that although it does suggest an approach of mapping OWL to ebRIM in order to improve the capabilities of ebXML RegRep, it does not propose mapping in the opposite direction. Which we did in order to overcome the main problem mentioned in the introduction (Section I) of this paper.

Gruber defined ontology as “a formal specification of a shared conceptualization” [21]. Conforming to the proposed ontology definition, we implemented ebRIM ontology as a formal specification of ebXML Registry Information Model.

The ebRIM ontology represents a non-domain-specific metadata ontology. Its main purpose is to serve as meta-metadata ontology for other domain-specific metadata ontologies like: Dublin Core, ISO 82045, ISO 19115 and VRA Core. The ebRIM ontology can be considered a semantic basis for mapping other domain-specific metadata ontologies. The solution we propose in this section is implemented with the aim of enabling unified, document type independent SPARQL [22] based reasoning over heterogeneous metadata sources; at the same time leveraging the contained knowledge, as efficiently as possible.

A. Implementation of ebRIM Ontology

In this subsection we suggest one approach, based on ebRIM specification, for solving the main problem of this research.

According to previously mentioned ebRIM specification [23] we implemented unique metadata ontology in order to achieve semantic reasoning over various types of documents and their metadata. In this section the implementation of ebRIM ontology will be discussed in more detail, focusing mainly on the implementation strategies we used for creating the ontology.

The ebRIM ontology is implemented across several key layers, with each layer representing its fundamental concepts: core, classification, association and provenance. More specifically, ontology consists of four sub-ontologies: (1) *Core*, (2) *Classification*, (3) *Association*, and (4) *Provenance information model* ontology. (1) *Core information model* ontology is the main sub-ontology which defines core metadata classes, including the common base classes and relevant properties. (2) *Classification information model* sub-ontology provides a semantic means for the introduction of taxonomies amongst the aforementioned resources. On the other hand, (3) *Association information model* sub-ontology defines elements (classes and relevant properties) which in conjunction with Classification sub-ontology enable a semantically richer way of expressing many-to-many relationships between resources. Whereas, (4) *Provenance information model* sub-ontology represents a typical top-level ontology, made up of classes which enable description of provenance (source information) for the concepts proposed in Core ontology.

Figure 1 shows the ebRIM ontology class hierarchy, focusing on its affiliation to the key sub-ontologies by highlighting the graph nodes. Highlights, in a form of colored circles on the upper edge of each graph node, indicate the affiliation of the class to the corresponding sub-ontology. As previously mentioned, ebRIM ontology consists of four sub-ontologies: Core, Classification, Association and Provenance information model ontology, with each sub-ontology representing a semantic counterpart of a matching model defined by ebRIM specification; which means, the proposed ontology, notwithstanding minor changes, is largely based and highly influenced upon the modeling patterns of ebXML RIM.

Core information model ontology classes, marked with a bright shade of red (Fig. 1), are the following (given in alphabetical order): *ExternalIdentifier*, *ExternalLink*, *ExtrinsicObject*, *Identifiable*, *ObjectRef*, *RegistryObject*, *RegistryPackage*, *RepositoryItem*, *Slot* and *VersionInfo*. The role of each class is elaborated below:

- *ExternalIdentifier* class facilitates RegistryObject identification with external identity information. ExternalIdentifier instance has an identificationScheme object property, referencing the ClassificationScheme instance to identify external identifier type.
- *ExternalLink* class is used to associate a registry object with arbitrary content residing either in or outside the registry.

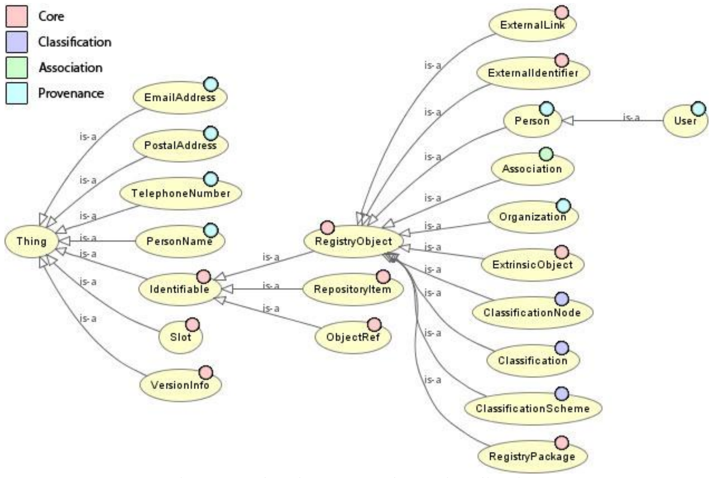

Figure 1. The ebRIM ontology visualization

- *ExtrinsicObject* class provides another strategy for ebRIM extensibility. ExtrinsicObject instances represent a repository item metadata of non-intrinsically known type, i.e. whose type is not known to the registry. It is a primary metadata class for a RepositoryItem.
- *Identifiable* class is a super class for the most classes within the ebRIM. Information model classes, whose instances require a unique identifier, are derived from the Identifiable class.
- *ObjectRef* class models a reference to a registry object. An ObjectRef instance is used to reference a RegistryObject instance.
- *RegistryObject* class serves as a common base type for all metadata elements in ebRIM ontology. Additionally, it can be considered an extensible metadata container, since it is responsible for holding the metadata.
- *RegistryPackage* class allows semantically related RegistryObject instances to be grouped together in a RegistryPackage.
- *RepositoryItem* class represents a repository item, the owner of metadata. RepositoryItem instances can be considered as an abstract view of a document.
- *Slot* class provides a means for extensibility within the ebRIM, whereas Slot instances serve as extensible attributes for registry objects. They enable a dynamic way of adding arbitrary attributes to registry entries, making the information model flexible in modeling various metadata standards.
- *VersionInfo* class represents information about the specific version of RegistryObject.

Above mentioned Core sub-ontology classes represent a subset of classes proposed by Core information model specification, with the addition of RepositoryItem class, which represents the owner of metadata, i.e. an abstract view of a document. Although this class originally does not belong to ebRIM, the introduction of RepositoryItem plays an important role in the establishment of a connection with metadata, semantically modeled by RegistryObject class. This addition is aimed at facilitating the extensibility of meta-metadata ontology, as it allows an easy way of semantic linking with other, widely used document-based ontology models, which enables even more comprehensive semantic reasoning.

Classification information model ontology classes, marked with a bright shade of magenta (Fig. 1), are the following:

- *Classification* instance explicitly classifies a RegistryObject instance, by referencing a ClassificationNode instance, defined within a ClassificationScheme.
- *ClassificationNode* instance enables refinement of ClassificationScheme tree-like structures. Taxonomy trees are constructed by nesting ClassificationNode instances underneath a ClassificationScheme instance.
- *ClassificationScheme* class equips ebRIM with a means for associating hierarchical information in a form of domain-specific taxonomies to RegistryObjects. ClassificationScheme instances represent tree-like structures (taxonomy trees) for classification of RegistryObject instances.

Association information model consists of only one class, marked with a bright shade of green (Fig. 1), whose role is to provide a strategy for associating two RegistryObject instances. An Association instance represents a many-to-many relationship between two RegistryObject instances. Association instance has an associationType object property, pointing to the ClassificationScheme instance, used to identify its type linking it to a domain-specific taxonomy.

Provenance information model ontology classes, marked with a bright shade of blue (Fig. 1), are the following:

- *Person* instance represents a person or human being.
- *User* instances represent a user known to the registry.
- *Organization* instances provide general information about organizations, where each Organization instance may have a reference to its parent.
- *PostalAddress* class defines data type properties of a postal address.
- *EmailAddress* class defines data type properties of an email address.
- *TelephoneNumber* class defines data type properties of a telephone number.

In the next section, a case study of mapping Dublin Core metadata element set to ebRIM meta-metadata ontology is further elaborated.

V. Case Study: Mapping Dublin Core to ebRIM

After the implementation of ebRIM ontology, another step is to be taken in order for a DMS to fully leverage the benefits of using a semantic approach for metadata modeling. That step includes mapping domain-specific metadata ontologies to ebRIM ontology. In the case an appropriate ontological representation of a domain-specific metadata does not exist, mapping of element set to a domain-specific ontology precedes ebRIM ontology mapping. An example of such a case would be mapping of Akoma Ntoso [24], an XML schema for description of parliamentary, legislative and judiciary documents, to ebRIM metadata ontology. Creating an OWL ontology based on Akoma Ntoso metadata element set described in XML schema, in this case, precedes mapping between these two ontologies, Akoma Ntoso and ebRIM.

As a case study of our research we decided to use Dublin Core, a domain-specific metadata element set [25]. The reason we chose Dublin Core over a number of other domain-specific metadata alternatives is that it is equally used to describe web as well as physical resources, which is why it is so widely employed as a metadata standard.

The most common ontological representation of Dublin Core metadata element set available online is protégé-dc [26] proposed by the authors from Stanford University. Since it is mainly intended to provide the annotational information for other ontologies, the protégé-dc implementation suggests annotation properties for modeling metadata elements. However, due to the restrictions of annotation properties, such implementation does not provide enough means for efficient semantic reasoning. According to OWL Reference document [27], annotation properties provide no semantics in OWL DL, and therefore are completely ignored by the reasoner.

Considering the limitations of annotation properties in terms of the semantic reasoning, we did not find protégé-dc ontology to meet our requirements. For this reason, we implemented (Fig. 2) a unique Dublin Core metadata element set ontology using OWL constructs which enable efficient semantic reasoning.

As shown in Figure 2, there are two major classes on the same level of the hierarchy: *Metadata* and *Field*. Metadata class allows modeling Dublin Core metadata element set. Whereas the elements are modeled as Fields of which each Metadata instance consists. Class Field provides a super class for other classes representing the 15 core elements defined in Dublin Core specification: *Contributor*, *Coverage*, *Creator*, *Date*, *Description*, *Format*, *Identifier*, *Language*, *Publisher*, *Relation*, *Rights*, *Source*, *Subject*, *Title* and *Type*.

The relation between Metadata and Field individuals is modeled as an object property *hasField.* On the other hand, actual metadata content of each Field individual is linked with a data type property *textContent*.

Linking domain-specific metadata ontologies to ebRIM ontology is the final step in enabling semantic web-based reasoning over various metadata sources. In order to overcome disparities between Dublin Core and ebRIM ontologies, and define exact semantic rules for mapping different concepts, we introduced SWRL rules.

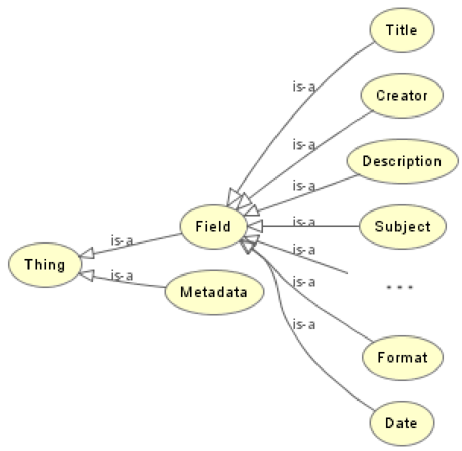

Figure 2. Dublin Core metadata element set ontology

Due to their expressiveness, we used SWRL (Semantic Web Rule Language) [28] rules in a form of Horn clauses, as a means for mapping concepts over various metadata sources. SWRL rules are combined with OWL knowledge base, thus enabling comprehensive semantic web-based reasoning using an extended set of OWL axioms.

Listing 1 shows SWRL rules used for mapping typical concepts between these two ontologies, where each rule represents a set of similar rules relevant for mapping a corresponding set of Dublin Core classes. The first set of rules applies to: *Contributor*, *Creator*, *Publisher*, *Subject*, *Date*, *Type*, *Format*, *Language*, *Coverage* and *Rights*; the second: *Title* and *Description*, the third: *Identifier*, whereas the last one provides a rule for mapping: *Source* and *Relation* Dublin Core ontology classes to ebRIM.

Mapping of other domain-specific metadata ontologies to ebRIM ontology may be achieved in a similar manner, following the very same concept defined in this section.

VI. Conclusion

In this paper we presented an approach for enabling unified, document type independent semantic web-based reasoning over various metadata sources in a DMS. The proposed solution is influenced by ebXML RegRep standard, rested on semantic web technologies. We implemented a non-domain-specific metadata ontology based on ebRIM specification. The main purpose of ebRIM ontology is to serve as meta-metadata ontology for other domain-specific ontologies. It is implemented with the aim of providing a comprehensive, yet rather concise semantic basis for mapping other domain-specific ontologies. As an example of domain-specific metadata ontology, without loss of generality, we used Dublin Core metadata element set. To overcome disparities between Dublin Core and ebRIM ontologies and define exact inferential rules for mapping different concepts, we used SWRL rules as a means for mapping concepts over various metadata sources. Due to its comprehensiveness and extensibility, other domain-specific metadata ontologies can be mapped to ebRIM ontology using the same concept, which proves ebRIM to be an ontology capable of dealing with a wide range of domain-specific metadata. At this point, ebRIM ontology does not support all aspects of ebRIM specification, which might be restricting in terms of modeling mostly document metadata. Information models such as: Service, Event and Cooperating registries, were not included in the current

```
1° RegistryObject(?regObj) ∧ Slot(?slot) ∧ Metadata(?metaData) ∧ Subject(?field) ∧ hasField(?metaData, ?field) ∧
   isMappedTo(?metaData, ?regObj) ∧ slots(?regObj, ?slot) ∧ textContent(?field, ?text) → slotName(?slot, "Subject") ∧
   slotValue(?slot, ?text)
2° RegistryObject(?regObj) ∧ Metadata(?metaData) ∧ Title(?field) ∧ hasField(?metaData, ?field) ∧ isMappedTo(?metaData,
   ?regObj) ∧ textContent(?field, ?text) → name(?regObj, ?text)
3° RegistryObject(?regObj) ∧ Metadata(?metaData) ∧ Identifier(?field) ∧ hasField(?metaData, ?field) ∧
   isMappedTo(?metaData, ?regObj) ∧ textContent(?field, ?text) → lid(?regObj, ?text)
4° ExternalLink(?extLink) ∧ RegistryObject(?regObj) ∧ Metadata(?metaData) ∧ Source(?field) ∧ externallyLinks(?regObj,
   ?assoc) ∧ isExternallyLinkedBy(?extLink, ?assoc) ∧ hasField(?metaData, ?field) ∧ isMappedTo(?metaData, ?regObj) ∧
   textContent(?field, ?text) → externalURI(?extLink, ?text) ∧ linkType(?extLink, "Source")
```

Listing 1. SWRL rule-based mapping between ontologies

version, since they were considered of less importance for document metadata modeling. However, they might be an important addition in extending ebRIM ontology towards metadata related to resources other than documents. We also plan some enhancements in the field of Provenance Information model ontology, which can be semantically enriched by incorporation of standardized top-level ontologies such as: FOAF, ORG, PROV-O and other ontologies for semantic handling of document related provenance information.

REFERENCES

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

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

[3] World Wide Web Consortium (W3C), (2012). OWL 2 Web Ontology Language Document Overview [online] Available at: http://www.w3.org/TR/owl2-overview/ [Accessed 20 Nov. 2014].

[4] Chen, X., Zhu, X., Zhang, X. and Du, D. “Geospatial data and services semantic share based on ebXML registry”, Second International Conference on Space Information Technology, 2007.

[5] Chen, N., Di, L., Yu, G., Gong, J. and Wei, Y., “Use of ebRIM-based CSW with sensor observation services for registry and discovery of remote-sensing observations”, Computers & Geosciences, 2009.

[6] Dogac, A., Laleci, G., Kabak, Y., et al., “Exploiting ebXML Registry Semantic Constructs for Handling Archetype Metadata in Healthcare Informatics”, International Journal of Metadata Semantics and Ontologies, 2006.

[7] Bechini, A., Tomasi, A. and Viotto, J., “Enabling ontology-based document classification and management in ebXML registries”, Proceedings of the 2008 ACM Symposium on Applied Computing (SAC), Fortaleza, Ceara, Brazil, 2008.

[8] OASIS ebXML Registry Webinar, (2005). [online] Available at: https://www.oasis-open.org/presentations/registry_webinar_05.ppt [Accessed 20 Nov. 2014].

[9] OASIS ebXML RegRep Version 4.0. Overview Document. (2012). [online] Available at: http://docs.oasis-open.org/regrep/regrep-core/v4.0/regrep-core-overview-v4.0.pdf [Accessed 20 Nov. 2014].

[10] Arsic, B., Đokic, M. and Stefanovic, N., “Mapping ebXML standards to ontology”, International Conference on Information Society and Technology (ICIST), 2014.

[11] Song, D., Liu, W., He, Y. and He, K., “Ontology Application in Software Component Registry to Achieve Semantic Interoperability”, Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC), 2005.

[12] Bechini, A., Tomasi, A. and Viotto, J., “Enabling ontology-based document classification and management in ebXML registries”, Proceedings of the ACM Symposium on Applied Computing (SAC), 2008.

[13] Bahaj, M. and Baroudi, S., "Integrating Ontologies Into ebXML Registries for Efficient Service Discovery", International Journal of Applied Engineering and Technology, 2014.

[14] Dogac, A., Kabak, Y. and Laleci, G.B., “Enriching ebXML Registries with OWL Ontologies for Efficient Service Discovery”, 14th International Workshop on Research Issues in Data Engineering (RIDE-WS-ECEG), Web Services for E-Commerce and E-Government Applications, Boston, MA, USA, 2004.

[15] Houbie, F. et al. “Semantic Metadata in Catalogue”, Ontology and Discovery Workshop, Lecture conducted from ESRIN, 2009.

[16] Dogac, A., Laleci, G., Kabak, Y., et al., “Exploiting ebXML Registry Semantic Constructs for Handling Archetype Metadata in Healthcare Informatics”, International Journal of Metadata Semantics and Ontologies, 2006.

[17] E. Chiu, ebXML Simplified: A Guide to the New Standard for Global E-Commerce. New York: John Wiley, 2002.

[18] OASIS RegRep Technical Committee [online] Available at: https://www.oasis-open.org/committees/regrep [Accessed 20 Nov. 2014].

[19] OASIS RegRep Wiki. (2014). [online] Available at: https://wiki.oasis-open.org/regrep/ [Accessed 20 Nov. 2014].

[20] Sounderpandian, J. and Sinha, T. E-business process management: Technologies and Solutions. Hershey: Idea Group Publishing, 2007.

[21] Gruber, T. R. (1992). What is an Ontology? [online] Knowledge Systems, Stanford University. Available at: http://www-ksl.stanford.edu/kst/what-is-an-ontology.html [Accessed 20 Nov. 2014].

[22] World Wide Web Consortium (W3C), (2013). SPARQL 1.1 Overview. [online] Available at: http://www.w3.org/TR/sparql11-overview/ [Accessed 20 Nov. 2014].

[23] OASIS ebXML RegRep Version 4.0. Registry Information Model (ebRIM). (2012). [online] Available at: http://docs.oasis-open.org/regrep/regrep-core/v4.0/os/regrep-core-rim-v4.0-os.pdf [Accessed 20 Nov. 2014].

[24] Akomantoso.org, (2014). Akoma Ntoso — Site. [online] Available at: http://www.akomantoso.org/ [Accessed 17 Feb. 2015].

[25] DublinCore, (2012). DCMI Metadata Terms. [online] Available at: http://dublincore.org/documents/dcmi-terms/ [Accessed 26 Nov. 2014].

[26] Protégé, (2008). Stanford DC OWL. [online] Available at: http://protege.stanford.edu/plugins/owl/dc/protege-dc.owl [Accessed 26 Nov. 2014].

[27] World Wide Web Consortium (W3C), (2009). OWL Web Ontology Language Reference. [online] Available at: http://www.w3.org/TR/owl-ref/ [Accessed 26 Nov. 2014].

[28] World Wide Web Consortium (W3C), (2004). SWRL: A Semantic Web Rule Language. [online] Available at: http://www.w3.org/Submission/SWRL/ [Accessed 26 Nov. 2014].