Semantically Enriched Obligation Management
An Approach for Improving the Handling of Obligations Represented in Contracts

Barbara Thönssen[1,2] and Jonas Lutz[1]
[1]*Institute for Information Systems, University of Applied Sciences and Arts Northwestern Switzerland FHNW, Riggenbachstr. 16, Olten, Switzerland*
[2]*Dipartmento di Matematica e Informatica, University of Camerino, Via Madonna delle Carceri 9, Camerino (MC), Italy*

Keywords: Enterprise Architecture, Enterprise Ontology, Risk Management, Contract Management, Information Extraction, Obligation Management.

Abstract: Contract Management becomes increasingly important for companies and public administrations alike. Obligations and liabilities are described in contract clauses that are often buried in documents of a hundred pages and more. Although commercial Contract Management Systems (CMS) are available, with a few exceptions relevant information has to be extracted manually which is time consuming and error prone. But even if information extraction is automated and contracts are managed using a CMS, dealing with obligations is still a challenge. Whereas the CMSs deal well with time triggered obligations like periodical payments by setting up corresponding workflows, they fail to trigger obligations based on events, as this knowledge is out of the systems' scope. We introduce an approach to fill the gap as we relate information about the obligations managed in a CMS with background knowledge modelled in an ontology. The ontology is a formal representation of an enterprise architecture extended by top-level concepts. Motivating scenario for the approach is the contract management of a large company. For proof of concept a prototype has been developed.

1 INTRODUCTION

Contracting is a critical issue for enterprises and public administrations alike, as "All organizations depend upon their ability to make, manage, monitor and perform against their business commitments, or to structure and oversee those they receive from their trading partners" (IACCM 2007). To manage not only the sheer volume of contracts but their growing complexity, The International Association for Contract and Commercial Management (IACCM) believes that automation is fundamental and of critical importance. In a study that IAACM provided (IACCM 2007), benchmarks have shown that the benefits of Contract Management Software are significant for most organizations, despite reservations over the maturity and functionality of the available options. One software flaw is the weak support of obligation management. Even if a Contract Management System (CMS) is in place, clauses, terms, conditions, commitments and milestones are buried in unstructured text. Thus, a first step can be document segmentation and information extraction. In the Swiss national funded project DokLife contract documents are analysed and information like contract partners, contract date, jurisdiction, etc. is automatically extracted. Moreover, the documents are divided into disjoint sections for further analysis, e.g. classification and section specific information extraction. The original contract document, its segments and the extracted metadata are imported into a commercial CMS.

However, although with such an approach contract *document* management is improved, *obligation* management remains a challenge. Whereas time triggered obligations (e.g. sending a report at a certain time) can be supported by the workflow functionality of a Contract Management System, event triggered obligations cannot. The problem is that these events occur outside the CMS and cannot be foreseen. If for example a company goes bankrupt it is of vital interest to know whether a relationship exits with this company, what kind of relationship that is, whether the relationship is contracted, and whether obligations are to be met.

Our approach aims at providing the context for contract management by linking the extracted

Thönssen B. and Lutz J..
Semantically Enriched Obligation Management - An Approach for Improving the Handling of Obligations Represented in Contracts.
DOI: 10.5220/0004134700400048
In *Proceedings of the International Conference on Knowledge Management and Information Sharing* (KMIS-2012), pages 40-48
ISBN: 978-989-8565-31-0

information, i.e. contract metadata to business objects described in an enterprise ontology. In the ontology (background) knowledge about business objects and business events are made explicit. Hence, this knowledge can be used to automatically assess business events, for example reported in external sources of commercial information or newspaper reports, and to identify affected obligations for which the respective contract(s) can be retrieved.

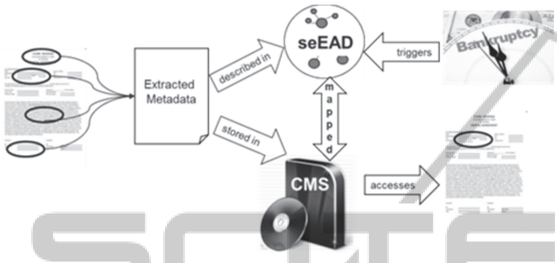

Figure 1: Obligation management overview.

Figure 1 gives an overview of the approach: information is automatically extracted from contract documents, the metadata is stored in the relational database of an CMS and mapped to the respective concepts in a ontology, which is a semantically enriched Enterprise Architecture Description. A business event (e.g. bankruptcy) triggers the search for affected contracts in the ontology and if found, the related documents can be accessed via the CMS.

The remaining paper is structured as follows: Section two describes the motivating scenario for semantically enriched obligation management. In section three background and related work is presented. Section four introduces the three pillars of our approach and in section five we give proof of concept by describing an implemented prototype. We conclude in section six and give an outlook on future work.

2 MOTIVATING SCENARIO

“Contracts exist because separate legal entities that decide to form a relationship with each other need some formal record of understanding regarding the nature of the relationship and their respective rights and obligations“ (IACCM 2007). Motivating scenario for our approach is a large machine manufacturer with several regional subsidiaries all over the world. The enterprise as a whole as well as each subsidiary has hundreds of contracts with suppliers, customers, sub-contractors, etc. Contract Management is considered one of the top ten risks an enterprise must face (Grosse-Ruyken & Wagner 2011). Therefore, all contracts are managed with a CMS. The company’s own contracts are created with the CMS, but the biggest number by far is issued externally and thus paper copies are scanned and then stored in the CMS. Many of the contracts are voluminous and identification of relevant information, such as obligations and conditions is error prone, cumbersome and time consuming. However, if this metadata is entered the CMS provides simple workflow functionality to trigger time related obligations like periodic payments (to make or to receive), submissions of reports, etc.

Besides managing contracts with a CMS, the machine manufacturer mandated an information provider for monitoring news about business partners, competitors, suppliers, etc. A risk manager is in charge of evaluating the news with respect to their impact on the company. This task is also time-consuming as well as time-critical and error-prone (with respect to completeness of evaluation).

To improve contract capturing, metadata generation and obligation management, the tasks should be automated.

3 BACKGROUND AND RELATED WORK

Although the importance of contract management is well-known in the economy, little scientific research has been done on that topic. Contract management has been addressed in the European integrated project TrustCoM (Wilson, 2007), that aims to develop a framework for trust, security and contract management in dynamic Virtual Organisations (Dimitrakos et al., 2004). “Trust between VO [Virtual Organisations] members can be supported by each being transparently aware of the obligations and performance of others, so that business risks are both mitigated, and monitorable” (Wilson et al., 2006). There are two main differences to our approach: Firstly, TrustCoM obligation (policies) are modelled in the form of Event-Condition-Action rules which define how the VO should adapt to failures, changes in requirements, security events, etc. Secondly, TrustCoM starts from contract creation based on (structured) templates. We consider that ideal but oversimplified. Many companies already have templates, approved by their legal department, which they are obliged to use. Even so, many commitments remain imprecisely described in contracts, for example what efforts the mitigation duty requires or what point in time the obligation arises (Goetz and Scott, 1983). In our approach we address these issues as we start from unstructured contract documents and describe obligations semantically instead of transforming them.

A well-known approach for describing entities of an enterprise semantically is in terms of an ontology. Bertolazzi et al. (2001) analysed and compared existing ontologies, namely the Toronto Virtual Enterprise (TOVE) and The (Edinburgh) Enterprise Ontology (EO) with their own proposal for a Core Enterprise Ontology (CEO). Leppänen (2005) introduced a context-based enterprise ontology and referred in his contribution in addition to TOVE and EO to the REA Enterprise Information System. Most recently, (Thönssen and Wolff, 2010) introduced a ContextOntology for dealing better with change in enterprises.

Whereas the term 'ontology' emerged in the context of Artificial Intelligence and later in the World Wide Web, particularly of the Semantic Web (Dietz, 2006), roughly at the same time enterprise architecture' became generally known as a management topic in the end of the 1980ies (Matthes, 2011). Contrary to enterprise ontologies, which are concrete representations of (generalized) enterprise architectures developed to be re-used in enterprises (Uschold et al., 1998), adopted and enhanced to the enterprise's specific needs, Enterprise Architecture Frameworks (EAF) "provide the guidance and rules for developing, representing, and understanding architectures" (DoDAF, 2007, p 4).

Kang et al. (2010), Hinkelmann et al. (2010) and Thönssen (2010) suggested relating an enterprise ontology to an Enterprise Architecture Framework in order to increase the quality of the enterprise ontologies, for example with respect to completeness. Our approach is based on ArchiMate, a standard that not only provides guidance but comprises a lightweight and scalable modelling language for architecture descriptions (The Open Group, 2009). Since ArchiMate's representation language is based on the UML 2.0 notation for class diagrams the 'meaning' of the entities is not described. "A meaning […] is a specialized description that aims to clarify or stipulate a meaning. […] Typical examples of meaning descriptions are definitions, ontologies, paraphrases, subject descriptions, and tables of content" (The Open Group, 2009).

Despite the consent about using an ontology for describing meaning no agreement has been achieved yet on the appropriate level of formalization and the degree of formality (Fox and Grüninger, 1998). Fox and Grüninger (1998) regard an enterprise model as the *computational* representation of the structure, activities, processes, information, resources, people, behaviour, goals, and constraints of an enterprise. This means that the enterprise ontology should be represented in such a way that a machine can *process* it. Because ArchiMate's original representation language is not expressive enough to achieve that, a more formal representation is needed for our purpose, like for example RDF(S) or OWL (Yu, 2011).

Having enterprise objects represented in an ontology, for example the *business object* 'contract' and related context information like *business actor* 'contract partner' and *product* 'Service Level Agreement', allows the addition of background information to a contract. In other words, a contract document and its metadata can be stored in a CMS but the *business object* 'contract' and its context are stored in the enterprise ontology. That leads to the issue of relating databases to ontologies, a requirement that has been investigated from the very beginning of the semantic web (Spanos et al., 2011). Approaches to combine relational databases (RDB) and ontologies have become known as 'database to ontology mapping problem', or more generally characterized as 'object-relational impedance mismatch problem' (Spanos et al., 2011). The problem that has to be solved lies in the structural difference of the relational and object-oriented models. It has been studied from different points of view for various kinds of reason (Auer et al. 2010), like semantic annotation of dynamic web pages, heterogeneous database integration, mass generation of Semantic Web data or ontology learning (Spanos et al. 2011). Sahoo et al. (2009) distinguish between 'Automatic Mapping Generation' and 'Domain Semantics-driven Mapping Generation'. Whereas the first method directly maps RDB and RDF schemas, the latter considers "domain semantics that are often implicit or not captured at all in the RDB schema" (Sahoo et al., 2009, p 5). As the focus of our research here is not on the mapping, we refer to some excellent papers on the subject (e.g. Barrasa et al., (2004), Kontchakov et al., (2010) and Spanos et al., (2011)). We also refer to the W3C RDB2RDF Working Group who provides many publications on automatic mapping, for example a strategy for directly mapping relational data to RDF (Arenas et al., 2011), and a language specification (R2RML) to express customized mappings from relational databases to RDF datasets (Das et al., 2011).

For our approach, Direct Mapping (DM) is sufficient as it is performant, simple and easy to implement. Direct Mapping defines an RDF graph representation of the data in any relational database with a set of common datatypes (Arenas et al., 2011). The DM technique entails the transformation of relational database data and schema into an RDF graph which is called the direct graph. The relational tables are mapped to classes in an RDF vocabulary, and the attributes of the tables to properties in the vocabulary (Hert et al., 2001). This technique has been introduced by the RDB2RDF working group (W3C Working Group, 2011).

Our approach carries research further with respect to modelling the context of a contract in terms of an ontology that represents enterprise objects in way that is machine executable. Taking a standard of an enterprise architecture description as a basis, i.e. formalizing and enhancing ArchiMate, is a new approach to represent the background information needed for event based obligation management.

4 ENHANCING CONTRACT MANAGEMENT

Our approach comprises three parts: contract segmentation and classification, metadata creation, and obligation monitoring. Because of the limited space we focus on obligation management and only briefly introduce the other two parts; they are simply pre-requisites but our approach would work as well with manually entered metadata.

4.1 Automatic Contract Segmentation and Classification

In order to facilitate the problem of metadata creation especially for large contract documents a contract is automatically split into disjoint segments. Paragraph numbers and titles are recognized and the separated parts are stored for further processing.

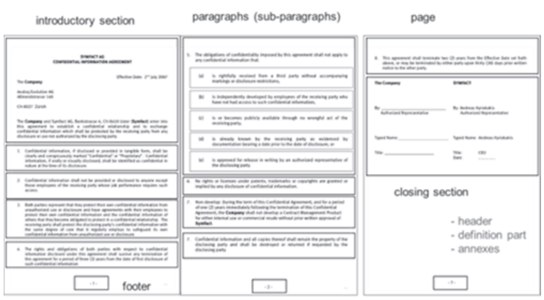

Figure 2: Contract segments.

Figure 2 shows the different parts of a contract document.

Contract segmentation is implemented in a commercial third party product for layout analysis to identify paragraph numbering based on regular expressions. After that, each of the paragraphs is classified with respect to the obligation type, e.g. *finance*, *report, notification*. Classification is done by a Support-Vector-Machine (libSVM, cf. http://www.csie.ntu.edu.tw/~cjlin/libsvm/). Paragraphs that are not numbered, like introduction and closing section or header and footer, are omitted as it is unlikely that they describe obligations.

4.2 Semi-automatic Metadata Creation

For the contract as a whole as well as for its paragraphs metadata are automatically extracted, e.g. contract partner, contract beginning, contract end, applicable law. For information extraction GATE (an open source solution for text processing cf. http://gate.ac.uk/overview.html) is used and some JAVA web-services. In addition to metadata for the whole contract particular metadata is created for single paragraphs, e.g. obligation type, trigger, dates and conditions. For this we also use regular expressions to extract due dates, conditions and triggers (time, period or event).

The analysed contract document, its paragraphs plus the created metadata are stored in an XML-file for further editing. In the DokLife research project for example, which is funded by the Federal Office for Professional Education and Technology OPET (Project KTI Nr. 10902.2 PFES-ES des Bundesamtes für Berufsbildung und Technologie) the file is imported into a commercial CMS (cf. to the project's Web-Site URL: http://www.doklife.ch for more information). Within the CMS for each obligation the text-segment and its metadata are displayed and the user can accept or correct it.

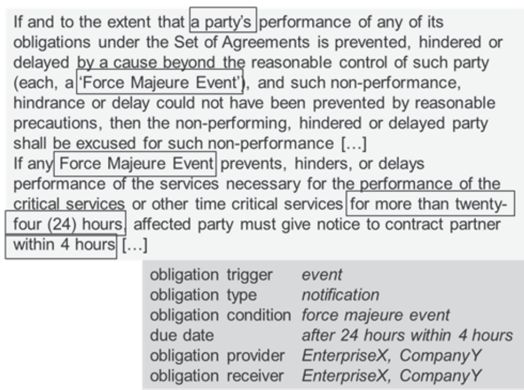

Figure 3: Paragraph example with Metadata.

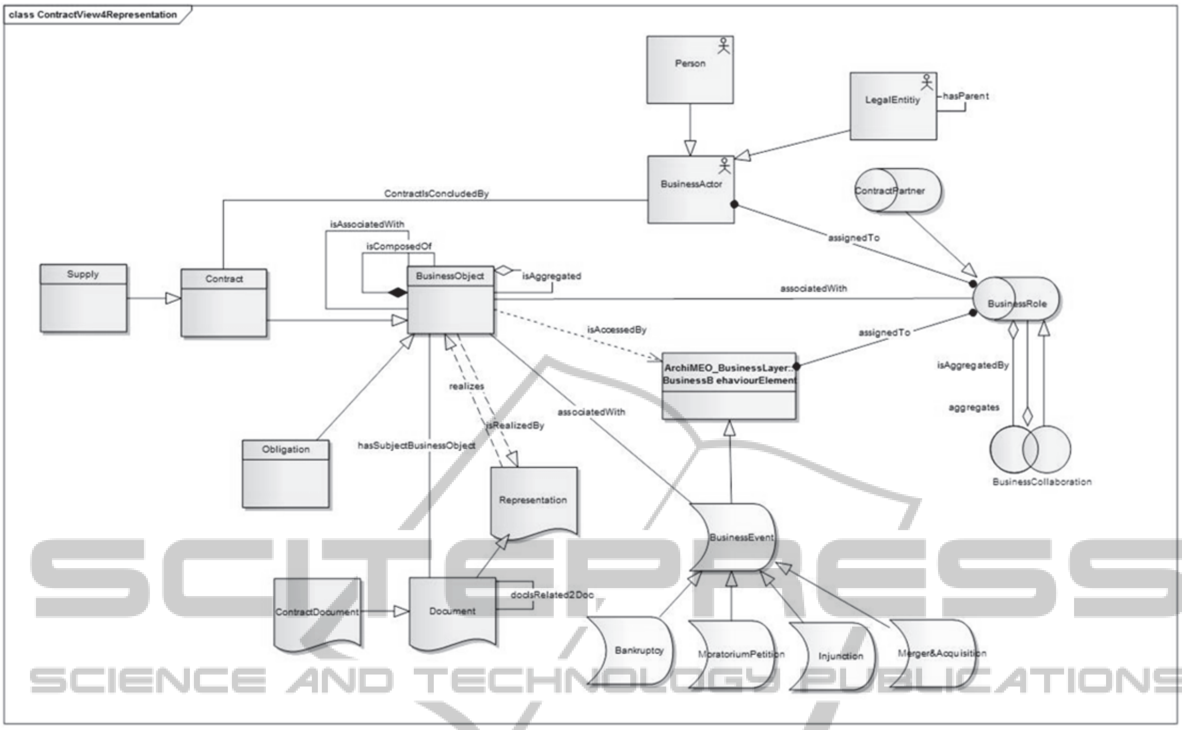

Figure 4: Excerpt of ArchiMate concepts.

Figure 3 depicts an example of a simplified paragraph recording the obligation regarding force majeure events plus the automatically created metadata. The paragraph is only one from an average of about 120 paragraphs extracted from a single contract. The information extracted from the paragraph is indicated by a rectangle. As shown, some of the obligation's metadata can be extracted (obligation trigger, obligation condition, due date), obligation type is provided by the SVM, and obligation provider and receiver are created by resolving the information 'a party' (which can be the contractor, here Enterprise X, or the contractee, here Company Y).

Contract segmentation and classification, and information extraction build the basis for improving obligation management.

4.3 A Broader View on Contract Management

To improve obligation management a contract document is considered a representation that realizes a business object (The Open Group, 2009) and "A business object is defined as a unit of information that has relevance from a business perspective". The quotation is taken from the ArchiMate framework, standardized by the The Open Group (2009). We adopted that notion and consider a contract document a representation that realizes a contract business object and one or more obligation business objects. In ArchiMate, business objects are related to business behaviour elements, such as business events, to business roles and business actors. With that all context information necessary for dealing with event-based obligations, recorded in contract are met.

For our purpose we refined the existing ArchiMate concepts with respect to creating sub-concepts and additional relations.

Figure 4 depicts a simplified excerpt of concepts and relations, depicted in the ArchiMate notation, relevant for our approach. We added two `BusinessObjects` `Contract` and `Obligation` that are realized in the `ContractDocument` (which is a `Represent-ation`). A `BusinessObject` is associated to a `BusinessEvent;` `Bankruptcy,` `MoratoriumPetition,` `Injunction` and `Merger&Acquisition` are (instances of) business events. A `Contract` is concluded with a `BusinessActor` with the assigned `BusinessRole` of a `ContractPartner`.

Conceptually we add three top-level concepts (with its sub-concepts) `time,` `location` and `event`. With this we are able to model events, externally to an enterprise, for example `ActOfGod` (Hurricane, Tsunami, etc.) or `ManMade` (war, riots, etc.), and to relate it to `BusinessObjects` in order to infer obligations triggered by force majeure events.

Further development of ArchiMEO followed Fox & Grüninger (1997) methodology on elaborating competency questions.

Example of an informal Competency Question:

```
Given an environmental disaster
(earthquake, hurricane, tsunami,
etc.) and some constraints (location
or about a business relationship),
which business partners are
concerned?
```

From the Competency Questions content of ArchiMEO has been validated and further elaborated as shown in Figure 4.

The Competency Questions are then rewritten in a formal way using SPARQL (see chapter 5).

5 SERO

As a proof of concept we developed a prototype for Semantically EnRiched Obligation management (SERO). Pre-requisite for SERO is the enterprise ontology and the results of contract segmentation and metadata creation for ontology population. This can be done automatically or manually.

The enterprise ontology used in SERO fully comprises the ArchiMate entities but is enhanced where necessary, as suggested in the standard (The Open Group 2009), and detailed above. Since a formal representation is needed the ArchiMate language has been translated into OWL. The enterprise ontology is named ArchiMEO to indicate its roots in ArchiMate. ArchiMEO is modeled with Protégé in OWL 2 QL dialect (it is also available in RDFS) and actually comprises about 400 concepts and 600 relations.

Currently the ontology is populated with a sub-set of the metadata resulting from the automatic contract segmentation, obligation classification and information extraction, namely `Contract-Document`, `Contract`, `Obligation`, `BusinessActor`, `Business-Role` and `BusinessRelationship`.

In the next phase Direct Mapping technique will be implemented to relate the instances with the entities stored in the relational database of the CMS.

For obligation management the ontology is imported into a SESAME triple store. Querying of the ArchiMEO ontology is realized in the SERO prototype by a JAVA Swing Interface implementing the OpenRDF Sesame API. The Application connects to the Sesame Triple Store and uses SPARQL queries, as exemplarily illustrated in chapter 5.1, to receive the required information. To simulate real world events the prototype interface contains fields to specify type and location of the occurring events. The depicted query in 5.1 presents the data related to a specific Use Case, introduced below, namely 'Earthquake' and 'Fukushima'. Of course, in SERO the implemented queries are more generic, allowing the search for any location and event stored in the Ontology. That is, via a graphical user interface one can ask for an event to search for affected business partners and if any contracted obligations. The resulting Information is added to JAVA objects and lists of objects such as `ContractDocument`, `Contract`, `Obligat-ion` and `BusinessActor` are displayed as tangent contracts by the event in the GUI.

To illustrate our approach for improving the handling of obligations represented in contracts, in the following two use cases examples are provided, based on the motivating scenario given above.

5.1 Use Case 1: Environmental Disaster

Initial Situation: In a newspaper an environmental disaster is reported and the machine manufacturer – as global player – wants to know if business partners are affected and if so, if obligations are due.

Analysis procedure: News, e.g. published on a newspaper web-site, are analysed using the same text analysis methods used for analysing the contract documents. For simplification in the prototype, such information is entered via a Graphical User Interface (GUI); in use case 1 this could be 'earthquake' and 'Fukushima'. Since in ArchiMEO events, e.g. earthquake, tsunami, flood are modelled and also information about locations, e.g. part of countries as Fukushima or Tōhoku are stored, knowledge about the type of the event (here: act of god) and the location (here: Japan) can be inferred. Based on the address of a business partner's production plants, headquarter, etc. it can easily be found out if he or she is affected or not. The query can be refined with respect to the closeness of a partner's location to the site of the disaster by exploiting geographical data for distance determination.

If a business partner is affected the kind of existing business relations can be inferred, respectively, what role that partner plays, e.g. supplier or consultant. Again, this information can be used to refine the query, as for example it would be more important for the manufacturer if the production plant of a supplier were in the epicentre of the earthquake than the headquarters of a consultant.

To find out whether obligations are due, the business object `obligation` is queried for obligation receiver and obligation provider. As depicted in Figure 3 each party has the duty notify the contract partner if a force majeure event affects its performance. With our approach the business partner affected by the earthquake in Fukushima is automatically identified, his or her obligations can be checked and, as an act of god like an earthquake is a force majeure event, this obligation can be identified also automatically. As result a warning can be issued and the contract document in which the obligation is represented can be retrieved from the CMS.

Example of the query in SPARQL:

```
PREFIX eo:<http://ch.fhnw.eo#>
PREFIX rdf:<http://www.w3.org/1999/
02/22-rdf-syntax-ns#>
PREFIX top:<http://ch.fhnw.top#>
PREFIX rdfs:<http://www.w3.org/2000/
01/rdf-schema#>

SELECT ?businessActor ?eventType
?eventParent ?obligation ?role ?location
?partCountry ?locationCountry ?contract
WHERE {
Eo:Earthquake rdf:type ?eventType.
?eventType rdfs:subClassOf ?eventParent.
{ ?businessActor
eo:businessActorIsSituatedInLocation
?locationCountry.
?locationCountry rdf:type top:Country.
} UNION { ?businessActor
eo:businessActorIsSituatedInLocation
?location
{ { ?location top:cityIsLocatedInCountry
?locationCountry } UNION {
?location
top:cityIsLocatedInPartOfCountry
?partCountry.
?partCountry
top:partOfCountryBelongsToCountry
?locationCountry. } }
}
?businessActor
eo:businessActorHasAssignedBusinessRole
?role.
?businessActor
eo:ibusinessActorPerformsObligation
?obligation.
?obligation
eo:obligationIsAgreedInContract
?contract.

FILTER(?location = eo:Fukushima||
?partCountry = eo:Fukushima ||
?locationCountry = eo:Fukushima)
}
```

5.2 Use Case 2: Bankruptcy and Dislocation

Initial situation: An information service provider gives notice that a company filed bankruptcy in one country but at the same time opens a new production plant in another country. The machine manufacturer wants to know what consequences this has for him.

Analysis procedure: The provided information is analysed and ArchiMEO is queried for obligations due in case of bankruptcy for that very business partner. Whereas this part of the query is easy to execute, finding out obligations related to the production site is not. In SERO it can be done by inferring information from the location. Assume the production site has been in Singapore and is now moved to Vietnam; the manufacturer is located in Switzerland. Since Switzerland has a free trade agreement with Singapore but not with Vietnam, the reason for the supply contract is not valid any more. In ArchiMEO legal bases for contracts can be described and thus, in use case 2 not only the obligation to notify the machine manufacturer about the business event could be identified automatically but also contracts affected by a change of legal pre-requisites.

As a side effect, records management for contract documents can be improved as well. If a contract partner goes bankrupt all contracts become invalid. The documents' retention period starts by then which can be triggered automatically based on ArchiMEO.

With our approach we can improve the handling of obligations represented in contract documents by providing the missing link to external business events (e.g. bankruptcy) or force majeure events (e.g. earthquake). We can automatically identify affected business partners, business relations, business roles inferring the enterprise ontology and trigger event-based obligations represented in contract documents which can be stored in a Contract Management System.

Our approach has been presented and reviewed by two focus groups, namely the user group of a commercial CMS and the consortium of the APPRIS project. APPRIS (Advanced Procurement Performance and Risk Indicator System) is a Swiss national funded research project (KTI-Nr. 12102.1 PFES-ES) with the goal of integrating risk, procurement and knowledge management into one early warning system.

6 CONCLUSIONS AND FURTHER WORK

"According to 42% of enterprises in a new study the top driver for improvements in the management of contracts is the pressure to better assess and mitigate risks" (Anon, 2007). With our approach, a first step in this direction is taken, as the risk of overseeing obligations and contracts affected by events (business, acts of god or man-made) can be minimized.

With the help of the SERO prototype the feasibility can be (and has been) demonstrated that obligations and contracts, triggered by business events, can be identified automatically and thus, contract management can be improved significantly.

Our approach will be used and further developed in the APPRIS project. The functionality of SERO will be used to detect warning signals based on risk indicators related to events in order to address the contract management risk.

Technologically, SERO will be improved with respect to instance management. Whereas at present the ontology is not yet physically linked to the contract documents stored in a Contract Management System, in the next version of SERO instances of ArchiMEO will be related to entities of of a CMS's database via Direct Mapping.

Currently, all background information is fully stored in ArchiMEO. In the future, Linked Open Data sources like GeoNames will be integrated. The GeoNames geographical database for example covers all countries, and contains over eight million place names (Wick n.d.). Also promising is the Open Government Data initiative in Switzerland (opendata.ch) with respect to integrating business administration issues like laws and regulations. Research will focus on how these sources can be truly integrated into ArchiMEO instead of simply mashing them up and displaying them alongside each other (Bizer et al., 2009).

REFERENCES

Anon, 2007. Contract Management: Optimizing Revenues and Capturing Savings. *Contract Lifecycle Management and the CFO*. Available at: http://findarticles.com/p/articles/mi_pwwi/is_200705/ai_n19058810/ [Accessed April 19, 2012].

Arenas, M. et al., 2011. A Direct Mapping of Relational Data to RDF. *W3C Working Draft 20 September 2011*, (September 2011), pp.1-11.

Auer, S. et al., 2010. Use Cases and Requirements for Mapping Relational Databases to RDF. *W3C Working Draft 2 June 2010*, (June), pp.1-18. Available at: http://www.w3.org/2001/sw/rdb2rdf/use-cases/ [Accessed January 26, 2012].

Bertolazzi, P. et al., 2001. An Approach to the Definition of a Core Enterprise Ontology : CEO. In *International Workshop on Open Enterprise Solutions: Systems, Experiences, and Organizations - OES-SEO 2001*. Rome, pp. 104-115.

Bizer, C., Heath, T. & Berners-Lee, T., 2009. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 5(3), pp.1-22.

Dimitrakos, T., Golby, D. & Kearney, P., 2004. Towards a Trust and Contract Management Framework for dynamic Virtual Organisations. In *Proceedings of the eChallenges Conference (eChallenges 2004)*. Vienna, Austria.

DoDAF, 2007. DoD Architecture Framework Volume I : Definitions and Guidelines. *Architecture*, I(April 2007), pp.1-46.

Fox, M. S. & Grüninger, M., 1998. Enterprise Modeling. *AI Magazine*, 19(3), pp.109-121.

Fox, M. S. & Grüninger, M., 1997. On Ontologies And Enterprise Modelling. In *International Conference on Enterprise Integration Modelling Technology 97*. Springer. Available at: http://www.eil.utoronto.ca/enterprise-modelling/papers/fox-eimt97.pdf.

Goetz, C. J. & Scott, R. E., 1983. Mitigation Principle: Toward a General Theory of Contractual Obligation. *Virginia Law Review*, 69(6). Available at: https://litigation-essentials.lexisnexis.com/webcd/app?action=DocumentDisplay&crawlid=1&srctype=smi&srcid=3B15&doctype=cite&docid=69+Va.+L.+Rev.+967&key=2c499ddc896515ef43456ba87a993188.

Grosse-Ruyken, P. T. & Wagner, S. M., 2011. *APPRIS Project Report*, Zürich.

Hert, M., Reif, G. & Gall, H. C., 2001. A Comparison of RDB-to-RDF Mapping Languages. In *Proceedings of the 7th International Conference on Semantic Systems*. Graz, Austria, pp. 25-32. Available at: http://dl.acm.org/ft_gateway.cfm?id=2063522&ftid=1054499&dwn=1&CFID=81605256&CFTOKEN=14608054.

IACCM, 2007. *Contract Management Software. Market Sizing And Status Report*,

Leppänen, M., 2005. A Context-Based Enterprise Ontology. In G. Guizzardi & G. Wagner, eds. *Proceedings of the EDOC International Workshop on Vocabularies, Ontologies and Rules for the Enterprise (VORTE'05)*. Enschede, Netherlands: Springer Berlin, pp. 17-24.

Sahoo, S.S. et al., 2009. A Survey of Current Approaches for Mapping of Relational Databases to RDF. , pp.1-15.

Spanos, D.-E., Stavrou, P. & Mitrou, N., 2011. Bringing relational databases into the Semantic Web: A survey. *Semantic Web – Interoperability, Usability, Applicability*, 0, pp.1-41.

The Open Group, 2009. *ArchiMate 1.0 Specification*, Available at: http://www.ncbi.nlm.nih.gov/pubmed/

20087110.

Thönssen, B. & Wolff, D., 2010. A broader view on Context Models to support Business Process Agility. In S. Smolnik, F. Teuteberg, & O. Thomas, eds. *Semantic Technologies for Business and Information Systems Engineering: Concepts and Applications*.

W3C Working Group, 2011. RDB2RDF Implementations. Available at: http://www.w3.org/2001/sw/rdb2rdf/wiki/Implementations [Accessed January 25, 2012].

Wick, M., GeoNames. Available at: http://www.geonames.org/ [Accessed April 22, 2012].

Wilson, E.M., 2007. *TrustCoM Project Final Report V1*, Available at: ftp://ftp.cordis.europa.eu/pub/ist/docs/st/trustcom-finalreport_en.pdf.

Wilson, M., Arenas, A. & Schubert, L., 2006. *The TrustCoM Framework for trust , security and contract management of web services and the Grid - V2*, Available at: http://www.clrc.ac.uk/Activity/ACTIVITY=Publications;SECTION=225.

Yu, L., 2011. *Developer's Guide to the Semantic Web* online., A. Springer. Available at: <http://lib.myilibrary.com?ID=308236>