ISWC2014 Industry Track – Abstracts

Chairs:

Axel Polleres
Alexander Garcia
Richard Benjamins
These informal proceedings contain the Extended Abstracts accompanying the talks of the ISWC2014 Industry Track.

The Industry Track featured presentations both from younger companies focusing on semantic technologies and large enterprises, such as British Telecom, IBM, Oracle, and Siemens, just to name a few. With a record number of 39 submissions (7 of which were selected for full presentations and 23 for short lightning talks) in the industry track this year, the mix of presentations demonstrated the success and maturity of Semantic Technologies in a variety of industry- and business-relevant domains. The extended abstracts for the industry talks will also be published in a separate companion volume in the CEUR workshop proceedings series after the conference.

Apart from these talks, The Industry Track featured a plenary keynote on The Semantic Web in an Age of Open Data by Sir Nigel Shadbolt, Chairman and Co-Founder of the UK’s Open Data Institute.

Axel Polleres, Alexander Castro and Richard Benjamin, 22 October 2014

Copyright information: All copyrights of the abstracts published herein remain with the authors of the respective abstracts.
**Regular presentations:**

* Takahiro Kawamura. Deployment of Semantic Analysis to Call Center

* Ludovic Langevine and Paul Bone. The Logic of Insurance: an Ontology-Centric Pricing Application

* Parvathy Meenakshy and John Walker. Applying Semantic Web technologies in Product Information Management at NXP Semiconductors

* Tony Hammond and Michele Pasin. Linked data experience at Macmillan: Building discovery services for scientific and scholarly content on top of a semantic data model

* José Gutiérrez-Cuellar and Jose Manuel Gomez-Perez. HAVAS 18 Labs: A Knowledge Graph for Innovation in the Media Industry

* Osma Suominen, Sini Pessala, Joumi Tuominen, Mikko Lappalainen, Susanna Nykyri, Henri Ylikotila, Matias Frosterus and Eero Hyvonen. Deploying National Ontology Services: From ONKI to Finto


**Short (Pechakucha-style) presentations:**

* Tope Omitola, John Davies, Alistair Duke, Hugh Glaser and Nigel Shadbolt. Ontology-Based Linking of Social, Open, and Enterprise Data for Business Intelligence.

* Dong Liu, Eleni Mikroyannidi and Robert Lee. Integrating Semantic Web Technologies in the Architecture of BBC Knowledge & Learning Beta Online Pages.

* Paolo Bouquet, Giovanni Adinolfi, Lorenzo Zeni and Stefano Bortoli. SICRaS: a semantic big data platform for fighting tax evasion and supporting social policy making.


* Roberto Garcia and Nick Sincaglia. Semantic Web Technologies for User Generated Content and Digital Distribution Copyright Management.
* Frederik Malfait and Josephine Gough. RDF Implementation of Clinical Trial Data Standards.

* Jeff Pan, Yuan Ren, Gabriela Montiel and Zhe Wu. Fast In-Memory Reasoner for Oracle NoSQL Database EE: Uncover hidden relationships that exist in your enterprise data.

* Nico Lavarini and Silvia Melegari. Semantic Technology for Oil & Gas Business.

* Gokce Banu Laleci Erturkmen, Landen Bain and Ali Anil Sinaci. keyCRF: Using Semantic Metadata Registries to Populate an eCRF with EHR Data.


* Bernard Gorman, Jakub Marecek and Jia Yuan Yu. Traffic Management using RTEC in OWL 2 RL.

* Harry Halpin. The W3C Social Web Initiative

* Zhe Wu and Jay Banerjee. Efficient Application of Complex Graph Analytics on Very Large Real World RDF Datasets.

* Andreas Blumauer. SKOS as a Key Element in Enterprise Linked Data Strategies.


* Kwangssoo Kim, Eunju Lee, Soonhyun Kwon, Dong-Hwan Park and Seong-Il Jin. Health and Environment Monitoring Service for Solitary Seniors

* Ulli Waltinger. Smart Data Access: Semantic Web Technologies for Energy Diagnostics


* Pinar Gocebe, Oguz Dikenelli, Umut Kose and Juan F. Sequeda. Semantic Web based Container Monitoring System for the Transportation Industry.

* Bart van Leeuwen. iNowit, linked data as key element for innovation in emergency response. (talk only, no extended abstract)
Deployment of Semantic Analysis to Call Center

Takahiro Kawamura$^{1,2}$, Shinich Nagano$^1$ and Akihiko Ohsuga$^2$

$^1$ Corporate Research & Development Center, Toshiba Corp.
$^2$ Graduate School of Information Systems, University of Electro-Communications, Japan

1 Introduction

This paper presents an application of text data triplification for a business. Since this is an in-company application from a laboratory to a division, we cannot describe it as “a success story of business-relevant, industrial deployments of Semantic Web” in the CFP, although it will be useful as a case study.

Manufacturers have recently been endeavoring to deal effectively with a number of inquiries about product malfunctions, which are gathered at a call center. Nowadays, moreover, if the response to an inquiry is mishandled, users tend to be complainers in some cases. A bad reputation then spreads widely via social media, that is, “flaming” occurs, and may greatly affect sales of all the company’s products. Making the response more problematic for operators at the call center is the difficulty of distinguishing whether the malfunction that is the subject of the inquiry is caused by a user’s way of using the product or a problem that accidentally occurs in an individual product, or caused by a problem common to the design or production phase of a particular model. In the case that an operator considers the malfunction to be the user’s fault at the initial stage, and it subsequently turns out to be the manufacturer’s fault, a firestorm may occur that may lead to lawsuits. The Consumer Affairs Agency in Japan and several law firms warn that the initial response to an inquiry is especially important in general. However, since pernicious complainers exist, if the manufacturer always considers the inquiry to be the manufacturer’s fault, the cost will soar.

Therefore, we proposed a method of comparing semantically analyzed social media information and the inquiry content. For checking a product’s reputation on social media, we currently collect entries from several review sites by searching with specific keywords once a day. Then, a person in charge looks through the result list. In addition to the risk of the person overlooking an important entry, the problem of the current workflow is that a keyword search cannot find entity-relations. In the case of searching with keywords like “Product A”, “heated”, and “high temperature” from articles, many unrelated articles, which do not include a relation of $Product_A \xrightarrow{heated} high\_temperature$, will be found. To avoid these noises, if we search with the keywords within each sentence, only the limited sentences will be matched. Therefore, we propose a method of extracting entity (keyword) relations. For example, in the case of finding $A \rightarrow B \rightarrow C$ ($\rightarrow$ indicates a relation), $A \rightarrow B$ extracted from a sentence and $B \rightarrow C$ extracted from another sentence can be combined. In practice, we triplify entries about product malfunctions on social media, and convert them to a network of Linked Data in
Then, by searching for the content of the inquiry to the call center in the network, we confirm whether the same issue is currently spreading on social media and whether the inquiry is the tip of an iceberg. If there is a similar entry on social media, it is determined whether the inquiry content is a malfunction common to a model and, if so, the operator offers a polite explanation to the user and a notification is sent to a quality control (QC) section. Moreover, if the entry has causal links connecting to users’ dissatisfaction and discontent, a notification with high priority will be sent to the quality control section.

We, that is, our laboratory, brought the above-mentioned advantages to the attention of a division of our company, which manufactures and sells consumer electronics, and then received a research contract with a certain amount of R&D expenses.

2 Triplification of Social Media Information

To create a training dataset, firstly, we divided each sentence in the dataset into chunks of semantically consistent words by using Part of Speech (POS) analysis and syntactic analysis, and then manually labeled one of eight properties, namely, Subject, Action, Object, Location, Time, Modifier, Because, and Other, to each block. We then used conditional random fields (CRF) as a learning model, which is an undirected graphical model for predicting a label sequence for a sequence. The key point of the proposed method is that we also constructed approximately 250 annotation rules using the result of syntactic analysis and the predefined ontology. We then decided which of the CRF estimation and the rule decision should be adopted based on the estimation probability of CRF.

In addition, we determined identities of values (chunks), that is, entity linking, so that values of Subject, Object, etc. that have the same meaning refer to an identical node in the network, as much as possible. Finally, we unified the values that are determined to be identical to a node whose label is a typical value.

\[ \text{approx. ten million yen for a half year} \]
Figure 1 presents the flow when an inquiry is received at a call center. When the call center receives an inquiry from a user, an operator records the summary of the inquiry content as two or three sentences (call log). Each sentence is triplified in the format of \(<S_i, V_i, O_i>\), and then triples that have the same structure as the sentence are searched in the triple store. In detail, our SPARQL query to the triple store first finds \(S_s, V_s, O_s\) that have the same meaning as \(S_i, V_i, O_i\), respectively, and then confirms whether there is an ID node that has the values \(S_s, V_s, O_s\) with Subject, Action, Object properties, respectively. As a result, if a triple with the same structure as the inquiry content is found, we determine that the problem does not concern an individual product, but is common to a model. Moreover, the number of triples with the same structure is regarded as an amount of topics on social media. When querying the triple store to find \(S_s, V_s, O_s\), we also use the method of entity linking. Example graphs of social media entries and inquiry content are shown in Fig. 2.

4 Experiments on Triplification and Matching

4.1 Triplification of social media

In an experiment, we collected entries about a TV set manufactured and sold by our company from a well-known review site in Japan, and then conducted labeling, learning, and estimation with the method described in the previous section. The dataset is 197 sentences for three months, and evaluated with 10-fold cross-validation. Table 1 shows the combined result of the CRF estimation in the case of the estimation probability \(p > 0.6\) or the rule decision, otherwise. W. Ave., which is an average value according to the number of each property, indicates that the combined method we proposed achieved accuracy of 94.1%.
Table 1. Extraction accuracy (above) and Matching accuracy (below)

<table>
<thead>
<tr>
<th>%</th>
<th>SUBJECT</th>
<th>OBJECT</th>
<th>ACTION</th>
<th>LOCATION</th>
<th>TIME</th>
<th>MODIFIER</th>
<th>BECAUSE</th>
<th>W. Ave.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>85.7</td>
<td>88.8</td>
<td>96.9</td>
<td>63.6</td>
<td>100.0</td>
<td>88.2</td>
<td>100.0</td>
<td>94.1</td>
</tr>
<tr>
<td>Recall</td>
<td>100.0</td>
<td>92.7</td>
<td>95.3</td>
<td>48.7</td>
<td>57.9</td>
<td>91.3</td>
<td>100.0</td>
<td>94.1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>No match</th>
<th>Match</th>
</tr>
</thead>
<tbody>
<tr>
<td>No data</td>
<td>Triplification Error</td>
</tr>
<tr>
<td>9.1%</td>
<td>13.5%</td>
</tr>
<tr>
<td>88.2%</td>
<td>84.3%</td>
</tr>
</tbody>
</table>

The accuracy of the Location is lower than that of other properties because of the shortage of geographical names registered in the system. The low accuracy of the Time seems to be attributable to the difficulty of distinguishing it from the Modifier. We also confirmed that extraction of the causal relation is feasible, since the accuracy of the Because property is high.

The division to which we provided this result commented that the 94.1% extraction accuracy is satisfactory, but pointed out that on this occasion social media information was collected for a certain period and converted to a graph (Linked Data), and therefore the graph represents a snapshot. Opinions expressed on social media are continually changing from product release to malfunction discovery and manufacturers’ responses, and thus such time-series variations should be represented in the graph. In addition, users’ complaints are of varying strength, and thus they should be divided into multiple stages from a weak complaint to a strong complaint.

4.2 Matching between inquiry content and social media

In the experiment, we first extracted 220 call logs (summaries of inquiry contents described by operators) from 25,459 logs about our company’s TV sets for a month, September 2012. We then compared them with social media information that was triplified as described in 4.1. Finally, the matching results between the call log and part of social media were manually checked.

The fact that the precision of call logs to social media graph was about 90% indicates that checking the same entry on social media as a call log is possible. Since the recall was low, however, we found that it is difficult to deduce how widely the call log is spreading on social media from this result. The recall was low because there are several expressions that represent the same content, and the entity linking is insufficient to unify them.

The division commented that when an inquiry is received at a call center, it is not possible due to time constraint that an operator performs keyword search with appropriate keyword expansion, and find the same entry as the inquiry content on social media, but this system automated comparison between call logs and social media with entity relations. The comment also indicated that in future when the malfunction of a model is spreading on social media, an alert should be transmitted before receiving the call log.

5 Conclusions and Future Work

Future works include performance evaluations. We have developed the system and are in the trial phase. In the future, we intend to identify issues that may arise through the actual operation of the system, and further improve the system.
The Logic of Insurance: an Ontology-Centric Pricing Application

Ludovic Langevine\textsuperscript{1} and Paul Bone\textsuperscript{2}

\textsuperscript{1} Mission Critical IT, Brussels, Belgium
\texttt{llg@missioncriticalit.com}

\textsuperscript{2} Mission Critical Australia, Melbourne, Australia
\texttt{pbo@missioncriticalit.com}

\textbf{Abstract.} We have developed a new pricing and scoring application for a large insurance company’s car insurance products. The business logic of this application is entirely expressed in an OWL ontology and SWRL rules. The ontology and rules are not merely documentation or a specification of the application, they form the business logic of the application. In other words the application is \textit{ontology-centric}. The application efficiently generates quotes, calculates the pricing of contracts and determines the applicability of a product based on an acceptance policy.

We report on our experience developing and deploying this system, as well as integrating it with legacy systems and new applications using mainstream technologies. We will show the significant quality and agility benefits of an ontology-centric approach that also delivers the expected performances on typical enterprise hardware.

\textbf{Scope and context of the project}

In an insurance company, actuaries define how products can be configured, how they are priced and at which market segments they are aimed. They document these details as reports and spreadsheets. IT people use these as the specifications of the systems to be built, creating multiple scalable systems, such as: product configurators, pricing services, scoring systems and acceptance checkers. These systems can be used through multiple channels, e.g. a call center, a Web site, insurance brokers, or partner car retailers. In this scenario, the introduction of a new insurance product or changes to existing acceptance rules or other aspects of existing products can take months and is subject to interpretation errors.

The development of Web insurance comparators and faster customer turnover represent challenges that insurance companies have to address in order to compete effectively in the market. The time-to-market for new or altered insurance projects must be as small as possible, this means the delay between the actuaries’ reports and actual changes to IT applications must be reduced.

To meet this challenge, a French subsidiary of the UK-based insurer Aviva decided to seek help from an external expert consultancy - LiteHouse Advisory - who proposed them an innovative solution and architecture which allowed them
to externalize the business logic of its car insurance products into an ontology. The ontology drives the behavior of an application which implements the products’ definitions. This application is then used by other systems such as the customer-facing Web site, and the sales-support application used by the call center. Changes to insurance products are made by changing the ontology and rules which, after testing, are used directly by the production application. This reduces the time-to-market dramatically.

The external consultant led the selection process to implement this solution, and Mission Critical IT was chosen to support the development of the ontology, to provide the needed technical components for IT to leverage it and to assist the IT department in the integration with other systems. A key component of the application architecture is Mission Critical IT’s ODASE\textsuperscript{3} platform for ontology-centric development. The application went live on the 10th June 2014.

### Functionality

The application exposes three services:

- **Pricing:** execute the product policy defined by the actuaries to produce quotes (determine the eligible products and their prices, apply the default selection based on questions answered by the customer) or compute the yearly premiums of existing contracts on renewal. This pricing service includes a configuration service, as it returns the set of optional insurance features a customer may chose and the default choices.

- **Acceptance:** depending on the available data about a customer (including incomplete data when some of the questions have not been answered yet) the application executes the accepting policy to determine whether an insurance offer can be made to the client or whether their contract can be renewed.

- **Scoring:** depending on the data about a new offer or a change of an existing contract, this service computes a statistical score which is used by the backend to decide whether the customer should be asked for additional documentary evidence (e.g. a statement from the previous insurer in the case of a new customer).

### Ontology-centric development and integration

The concepts and properties of the domain are expressed in OWL. Examples of concepts are *Vehicle, Person, Guarantee* and *Premium*. The ontology contains 208 concepts and 598 properties. SWRL rules are used with the axioms to provide reasoning. Two examples of rules are: a single rule defines the premium including taxes as the sum of the premium without taxes and the applicable taxes; a set of rules define which optional guarantees can be offered to the customer for a given quote and which of these should be selected by default.

\textsuperscript{3}http://www.missioncriticalit.com/odase.html
Three groups contributed to the project: actuaries, acting as subject matter experts (SMEs), two Cobol developers. It is worth noting that the ontological approach was very well received by these developers as they wanted to upgrade their skills and leverage their deep knowledge of the business. The Cobol developers created the ontology with coaching from ontologists from Mission Critical IT. The ontology was created based on interviews with the SMEs and then refined using an iterative Scrum agile methodology with three-week sprints.

The ODASE platform includes a tool that generates a Java API from an ontology. It creates a Java class for each OWL concept and Java getters and setters for each OWL property. OWL subclass axioms are translated into up- and down-casting methods. This ontology API interacts with the RDF stores and the OWL/SWRL reasoner which enforces the business logic expressed in the ontology at runtime. This API provides a type-safe and familiar interface to the business logic for Java developers.

The application is pure Java and runs as a plain servlet in a Tomcat container. The ontology files (OWL and RDF) and the ontology API are packaged within the WAR artifact with the runtime libraries of the ODASE platform. Demo applications were also developed using the API, these were used by the actuaries to test and refine the ontology during its development.

The actuaries often work with statistical and spreadsheet software. Specific import tools have been developed to extract data from Microsoft Excel spreadsheet and translate them into RDF instances. More than 5,000 instances have been generated this way to provide the ontology with all the required parameters.

The servlet listens to requests with two protocols: HTTP REST requests for modern client applications (the company Web site and external Web-based insurance comparators) and MQ Series requests from applications running on the legacy AS/400 system (backoffice and sales-support application for the call center). The amount of hand-written code specific to these services is very small: less than 2,000 lines of Java code.

The REST requests use RDF/XML formatted data, which is already in the business model defined by the ontology. However the MQ Series requests use Cobol data structures in the model of the legacy applications. A semantic integration with the AS/400 legacy system has been developed through the generation of intermediation ontologies from the description of these Cobol data structures.

Performances

The use of runtime reasoners and interpreted rules for such complex computations and inference tasks was perceived as a major risk for this project. In our opinion, this skepticism is mostly due to the lack of visibility of the Semantic Web technologies. To mitigate this risk, performance tests were conducted

---

4 The Cobol developers were selected for their abstract reasoning and strong understanding of the business.
throughout the project, under the supervision of the client’s IT department. Finally two load-balanced Intel Xeon systems proved enough to cope with the load during the busiest business hours.

ODASE parallelization mechanisms have been used to leverage the multiple cores of the servers. The query-based nature of its reasoners allows good response-times: the most complex request is answered in less than 400 ms during peak times. Quotes can be generated in 250 ms. The application serves about 30,000 requests per day.

Benefits

Business — IT alignment Traditionally such a business has, at least, two separate views of the business model; one maintained by the business and one (or more) maintained for the sake of IT applications. This requires extra effort and problems will occur when the models are out-of-sync. Ontology-centric development allows a single model to be shared by business and IT.

Agility IT and business people now share the ontology and understand the same concepts. The delivery of a new pricing or a change to the acceptance rules can be achieved in a few hours. The delivery time of new products has also been reduced. Changes in the ontology are reflected in the generated Java API when necessary and IT can modify applications as necessary, which is much less effort than implementing new products directly in Java. With ontology-centric development changes like these have a dramatically reduced time-to-market. As a matter of fact in the month since deployment, two change requests have already been successfully applied.

Quality In the three months following deployment approximately 200 defects have been identified in the whole project. Only two of these were in the ontology (approx 1%). The majority of defects were in the AS/400 screens and integration layers, followed by the Java Web application. Using ontology-centric development dramatically reduces the risk of bugs occurring in or effecting the business logic, where subtle bugs might be very expensive (for example, by miscalculating the price of a contract). This high level of quality is made possible thanks to the externalization of the business logic in a high-level formal language: the application is built on top of an executable and testable specification, rather than being the result of human interpretation of large and often inconsistent informal specifications. Moreover, because the business logic is implemented declaratively, any given output of the application can be explained in business terms, pinpointing the set of axioms and rules which are responsible for the result using the ODASE platform’s declarative debugging tool. The maintenance burden of the ontology was so low that two of the three developers (two from IT and one Mission Critical IT ontologist) were reassigned to other teams.
Applying Semantic Web technologies in Product Information Management at NXP Semiconductors

Parvathy Meenakshy (parvathy.meenakshy@nxp.com),
John Walker (john.walker@semaku.com)
2014-09-21

Abstract

In the electronics industry, the ability to get accurate, timely product data in front of the customer is a key factor in the overall business process. In this paper we describe how NXP is making use of Semantic Web technology such as RDF and SPARQL to manage a product taxonomy for marketing purposes that forms the key navigation of the NXP website (http://www.nxp.com) and the next steps to extend this to create a domain model covering applications, technologies and other key entities that can be used to create rich user journeys through the content.

Introduction

The ease of finding the right product is paramount for the customer and crucial for sales, but at the same time a complex problem in an eCommerce scenario. Enabling the customer to easily find, compare and select the right product can reduce the overall time and therefore costs involved in the purchasing process. In the B2B electronic component industry this can involve the choice between literally thousands of candidates. We believe opening up access to the product data is a key enabler for this process. Whether to free the data from existing silos for use within the organization or to make the data available to third parties such as distributors and search engines.

In this paper we focus on the development and management of taxonomies and describe how NXP has deployed a solution based on SKOS for the management of the product taxonomy and a currently-under-development approach to provide an alternate view of the product catalog.

Product taxonomy

The NXP product taxonomy [1] provides a marketing-oriented categorization for the product catalog. The taxonomy consists of categories organized into a hierarchical structure used for navigation on the NXP website. The taxonomy is primarily function-oriented, but also incorporates market, application and technology based categorization of products. It is permitted for a category to have more than one parent, forming a polyhierarchy. The taxonomy is used to manually position products into one or more categories as required for marketing purposes. Additional content and documents can be linked to a category.
Legacy approach
The product taxonomy is managed in the enterprise Product Information Management (PIM) system where each category is assigned a numeric identifier. Placement of products in the taxonomy is also done in the PIM system. The taxonomy and product assignments are exported from the PIM system in a proprietary XML format and loaded into an XML database. The document management system is able to query the XML database using XQuery to lookup categories to which a document can be linked using the numeric identifier. The XML is also imported to the web content management system and used to generate the website. Due to end of life of the PIM system, it was necessary to find an alternative approach for the management of the taxonomy.

Linked data approach
Previously we had created a process to convert the existing XML to RDF/XML and load the result into a graph store where it can be merged and queried along with data from other sources. For this we had created a mapping to SKOS where each category has the type skos:Concept and the hierarchical relations are mapped to the skos:narrower and skos:broader properties.

A logical next step was to make the RDF as the source and manipulate the data directly in the graph store using SPARQL 1.1 Update. After considering various open-source and commercial tools we opted to use SKOSjs [2] with some minor NXP-specific customizations and configuration.

For placing products into the tree we developed a simple application using the Play Framework that allows a user to manage the links. The application makes use of stored queries in the graph store that can be exposed as an HTTP API. Initial bindings for variables in the query can be passed as parameters in the request URI.

To support existing applications the legacy XML format is generated by transforming an RDF/XML dump of the data in the store using XSLT.

The benefits of this approach are multiple. First and foremost we have minted URIs as globally unique identifiers for each product category that can be used to unambiguously refer to the resource. The flexibility of the RDF data model has allowed us to add additional information such as alternate and translated labels for categories without disrupting existing applications. Additionally being able to use the power of SPARQL to flexibly query the data as a graph has opened up new ways to analyze the data and do quality and consistency checks. Users have also responded positively to having several simpler and more focused editing tools as opposed to the complex, yet generic user interface of the previous PIM system. In future we expect to gain further benefits from this approach.
Solution selling

For NXP the customer is typically an electronics design engineer. A common user story is a user who searches online for solutions to their design challenges rather than for a specific named component. Therefore NXP desires to serve the customer better by developing a solution-oriented view of the portfolio orthogonal to the existing function-oriented product taxonomy.

The specification of a product usually includes an textual applications chapter that lists the end-equipments, market segment and solution area in which the component can be used. This content can be indexed by site search and only gives the users a basic keyword search functionality. It does not provide an option for customers to explore the content under different perspectives or filter the data under search facets. Lack of a unique identifier for these named entities restricts reuse and aids ambiguity in referring to this information in different document assets.

A revamp of this static list of information would enable more effective and efficient discovery and selection of components which fit to a particular application/solution area and open up new opportunities for cross-selling related products and telling compelling stories about a particular focus area by aggregating relevant content. It should be noted that the application information of products does not always fit into a strict hierarchical structure. Moreover this information is subjected to change due to constant flux in technology trends and introduction of new products which fit into latest applications. A technology which is flexible enough to cope with the changes in schema is of prominence in our case. Semantic Web and Linked Data technologies enable an incremental development of domain model contrary to the conventional relational database schema. By applying these to describe the knowledge domain we can resolve ambiguities and empower machines to access the information more easily.

As described above we have already built experience with converting the NXP product taxonomy to RDF using the SKOS vocabulary where it is now stored and managed natively as RDF in a graph store. Additionally we have developed a simple web application that allows products to be linked to the taxonomy. Based on this success we are extending our use of semantic technologies by developing a domain model for applications/solutions that will drive a rich user-focused experience. This was based on the lessons learned from using SKOS where the terms like broader and narrower are very generic, whereas we need to have more domain-specific terminology to better capture the exact meaning. The approach taken was to first analyze the existing application content to extract common terms followed by domain modeling undertaken with the relevant subject-matter experts. Based on this model we have made wireframe designs for the various pages about the ‘things’ in the model which are due to be implemented in the coming months. During development of the model we encountered difficulties to explain the benefits of domain-driven design over a “UX first” approach which can often deliver tangible results faster, but can give a brittle and inflexible design. Areas we wish to investigate further are automated tagging of products and content assets (documents, images, video) with the concepts from the model and the possibility for a curatorial/personalization layer.
The approach
The approach used is domain driven design rather than presentation and interaction of UX design. One of the challenges was to identify an entity as a concept, story about a concept or a web page about a concept. Another was the difficulty in defining the concept in a data element fashion. This exercise is not so intuitive as we are not dealing with concrete or tangible things. We tried to map instances of data to the model and came up with potential user journey to agree on the relations between concepts. While trying to describe the semantic relation of application information to products, the priority was to conform to existing standards. SKOS and Dublin Core standards are used to express the basic metadata. But the semantics of SKOS is not rich enough to capture the application information of our products. Even though Schema.org is a light weight standard, defining all the concepts in our model with ‘Product’ class was not desirable.

Based on the model, instances were added with unique identifier or URI and are persisted to a named graph in a graph store (http://dydra.com/). A browsable interface of this data is created using the Linked Data API specification, implemented by open source project (https://code.google.com/p/puelia-php/). The API layer is configured to support REST API calls to the graph store. This provides the results in the HTML page without writing complex SPARQL queries and in different output format like RDF/XML, JSON etc. The ability to browse the data as HTML enables us to validate the model with the relevant subject matter experts. Later we expect that the same API layer can be used during generation of the pages on the NXP website.

Possible future work

One of the areas where we need to research further is on the auto tagging methods. This is of interest as automating the linking of content assets to concepts would reduce the manual work involved.

Similarly we would like to automate the categorization of products into the taxonomy by somehow expressing rules or constraints for the ‘type’ of products that should appear in the category and use this to automatically populate the product lists. As large amounts of the product specification are already available as structured data this is certainly possible, the real challenge is to make the management of these rules usable for the target audience.

References
Linked data experience at Macmillan:
Building discovery services for
scientific and scholarly content on top of
a semantic data model

Tony Hammond and Michele Pasin

Macmillan Science and Education, The Macmillan Campus,
4 Crinan Street, London, N1 9XW, UK
{tony.hammond,michele.pasin}@macmillan.com
http://se.macmillan.com

Abstract. This paper presents recent work carried out at Macmillan
Science and Education in evolving a traditional XML-based, docu-
ment-centric enterprise publishing platform into a scalable, thing-centric and
RDF-based semantic architecture. Performance and robustness guaran-
tees required by our online products on the one hand, and the need to
support legacy architectures on the other, led us to develop a hybrid
infrastructure in which the data is modelled throughout in RDF but
is replicated and distributed between RDF and XML data stores for
efficient retrieval. A recently launched product – dynamic pages for sci-
entific subject terms – is briefly introduced as a result of this semantic
publishing architecture.

Keywords: ontology, OWL, RDF, science, semantic publishing, XML

1 Background

Macmillan Science and Education is a publisher of high impact scientific and
scholarly information and publishes journals, books, databases and other services
across the sciences and humanities. Publications include the multidisciplinary
journal Nature, the popular magazine Scientific American, domain specific titles
and society owned journals under the Nature Publishing Group and Palgrave
Macmillan Journals imprints, as well as ebooks on the Palgrave Connect portal.

Traditionally we have operated an XML publishing workflow with a docu-
ment archive of over 1m articles and an averaged daily publication rate in the
100s of articles. As a prelude to moving towards a richer discovery environment
in 2012 we began to experiment with linked data technologies and set up a pub-
lic query service at data.nature.com with RDF metadata describing a simple
graph-based model.

In 2013 we embarked on a major new initiative to develop a new publish-
ing platform for nature.com. We both extended and refined the linked data

1 We have since retired this service but will continue to make data snapshots available.
model to manage our content and the relations between content items. Recently released discovery products in 2014 using this linked data foundation include scientific subject terms as a new navigational means, as well as bidirectional linking between articles and related articles.

2 Infrastructure

We have established a linked data architecture at the core of our publishing workflow and build on a common metadata model defined by an OWL ontology (see Fig. 1). This data model provides a number of significant benefits over traditional approaches to managing data: it encourages adoption of a standard naming convention by enforcing a global naming policy; it provides a higher-level semantic plane for data integration operations; it allows for flexible schema management consistent with an agile approach to software development; and finally it facilitates a simple means of maintaining rich dataset descriptions by allowing us to partition the data space using named graphs.

![Fig. 1. Overview of main classes in the ontology.](image)

To realize this common data model our Digital Systems, Science and Scholarly division has developed the Content Hub as part of the ongoing new publishing platform programme. All our publishable content is aggregated within the Hub which presents as a simple logical repository. In practice, the data is distributed across multiple physical repositories. The ontology organizes the conceptual data model as well as managing the physical distribution of content within the Hub using XMP packets\(^2\) for asset descriptions.

Our two core capabilities in managing the Hub are content storage and content discovery. Structured content is maintained as XML document sets within a MarkLogic repository which provides us with powerful text search facilities.

By contrast, discovery metadata is modelled in RDF (and constrained using an OWL ontology). The discovery metadata is further enhanced by using RDF rule sets: object-oriented contracts for generating knowledge bases, and SPIN rules for inferencing and data validation.

3 Challenges

Initially we attempted to query the triplestore and deliver data through a generic, linked data API but increasingly were frustrated in meeting delivery expectations especially as query complexity mounted with multiple includes, specific orderings, facetting, and text searching requirements. We found that our implementation failed in two critical dimensions: performance and robustness. Typical result sets were being delivered in seconds or tens of seconds, whereas we were being tasked to achieve ~20 ms, some 2–3 orders of magnitude faster. Additionally we faced system challenges based around enterprise features such as security, transactions, and updates.

It soon became apparent that in order to better support our online products we needed an application-oriented API that more directly reflected the page data model. This led to our developing a hybrid system for storage and query of the data model.

The main principles we used for the API were that data should be represented as consumed, rather than as stored; that it should be provided in a single call and support common use cases in simple, obvious ways; that it should ensure a consistent speed of response for more complex queries; and that it should build on a foundation of standard, pragmatic REST using collections and items.

The data organization within the Hub is shown in Fig. 2. The data is modelled throughout in RDF but is now replicated and distributed between RDF and XML data stores. We have added semantic sections as RDF/XML includes within our XML documents. Retrievals are realized with XQuery, and augmented by in-memory key/value lookups, yielding acceptable API response times typically in the 10–100 ms range depending on complexity. RDF queries are currently restricted to the build time phase of data assembly with data enrichment and integration managed at the ETL layer using both SPARQL query/update together with SPIN rules. The API delivers JSON to a front end for rendering as HTML by querying with XQuery over RDF/XML includes in XML documents. These RDF/XML sections are subsequently exported into a triplestore for offline model validation and reporting.

In sum we use RDF (plus OWL) as an organizing principle for our discovery data but we have preconditioned the data storage and index layouts in XML for simple and efficient retrieval access.

4 Products

The new publishing platform is increasingly being used to deliver products that benefit from a large-scale dynamic integration of related content. The first of
these products is Subject Pages\(^3\), a new section on the nature.com site that allows users to browse content topically, rather than via the more usual journal paradigm. This product was launched in early June 2014, and generated more than 200,000 views in its first month.

Subject Pages automatically aggregates content from across the site based on the annotation of that content (by authors and editors) using a taxonomy of scientific subject terms developed in-house. The taxonomy includes subject terms of varying levels of specificity such as *Biological sciences* (top level), *Cancer* (level 2), or *B 2 cells* (level 7). In total there are more than 2750 subject terms, organised into a polyhierarchical tree using the SKOS vocabulary. Subject Pages provides mechanisms to find items of interest, either by searching or navigating the hierarchy, and organizing the articles displayed on the page based on their article type (e.g. research or news). Moreover, subject terms are used to drive custom ads, jobs posts and events information which match the main topic of the page.

### 5 Future Work

Future aims are threefold: 1) to grow the data model with additional things and relations as new product requirements arise; 2) to open up the user query palette to more fully exploit the graph structure while maintaining an acceptable API responsiveness; and 3) to create an extended mindshare and understanding throughout the company in the value of building and maintaining the discovery graph as a core enterprise asset.

\(^3\) [http://www.nature.com/subjects](http://www.nature.com/subjects)
HAVAS 18 Labs: An Enterprise Knowledge Graph for Innovation in the Media Industry

José Gutiérrez-Cuéllar, José Manuel Gómez-Pérez, Boris Villazón, Nuria García, Aleix Garrido

1 HAVAS Media Group, Paseo de la Castellana 259 C planta 30
28046 Madrid, Spain
jose.gutierrez@havasmg.com

2 iSOCO – Intelligent Software Components S.A., Av. Del Partenón 10, oficina 1.3A
Campo de las Naciones 28042 Madrid, Spain
{jmgomez,bvillazon,ngarcia,agarrido}@isoco.com

Abstract. 18 months may be a short time in absolute terms but when referring to entrepreneurship, 18 months is usually a figure associated to life expectancy, with 80% technology startups crashing and burning in that period. HAVAS has launched 18 Innovation Labs, a global initiative aiming to identify startups in the intersection of technology and media, in order to co-create new ways to revolutionize the media and entertainment industry. With iSOCO’s support HAVAS seeks to interconnect startups, innovators, technology trends, other companies and universities worldwide in one of the first applications of web-scale knowledge graph principles to the enterprise world and media. The resulting enterprise knowledge graph will support analytics and strategic decision-making for the incorporation of such talent within their first 18 months life span. In this talk we describe the 18 Labs initiative, challenges, and business expectations and how semantic technologies are key for realizing this vision by extracting startup information from online sources, structuring and enriching it into an actionable, self-sustainable semantic dataset, and providing media businesses with strategic knowledge about the most trending innovations.

Keywords: Enterprise knowledge graph, acquisition, integration, consumption.

1 Introduction

The communication between brands and consumers is set to explode. Product features are no longer the key to sales and the combination of both personal and collective benefits is becoming an increasingly crucial aspect. As a matter of fact, brands providing such value achieve higher impact and consequently derive clearer economic benefits. On the other hand, millennials [3] are taking over; inducing a dramatic change in the way consumers and brands engage and what channels and technologies are required to enable the process. As a result traditional boundaries within the media industry are being stretched and new ideas, inventions, and technologies are needed to keep up with the challenges raised by the increasing demands of this data-intensive, in-time, personalized, and thriving market.

Thus, it is necessary to leverage advances in the area by stimulating a collaboration ecosystem between the different players. Inspiring examples include the adoption by
Tesla Motors of the open patents policy, whereby Tesla shares their innovation in regards to electric cars openly via the internet. In return, Tesla expects the industry to further evolve the electric car and dynamize the market. In the media industry a paradigmatic case of this ‘better together’ approach is HAVAS 18 Innovation Labs, deployed at strategic locations around the world. One of such locations is the Silwood research center in Santa Monica, co-created with Orange, which focuses on the convergence between technology, data science, content and media. 18 Innovation Labs seeks to connect a great mix of local talent over the sites, involving innovators, universities, start-ups and technology trends to co-create initiatives relevant now and in the mid-term for both HAVAS and their clients to stay one step ahead.

With the help of iSOCO, their partner in semantic technologies, HAVAS is creating an enterprise knowledge graph and information platform that aggregates all the available knowledge about technology startups worldwide and makes it available for exploitation by media business strategists through a single entry point. To the best of our knowledge this is one of the first applications of knowledge graph principles in the enterprise world, and the first in the media industry, after internet search giants Google, Yahoo, and Bing coined the term at web-scale, each with their own implementations. Related initiatives include domain-specific efforts like, social graphs like Facebook, and reference knowledge graphs like DBpedia and Freebase.

2 The HAVAS 18 Knowledge Graph

In a way analogous to the abovementioned initiatives, the main objective of HAVAS 18’s knowledge graph is to enable knowledge-based services for search, discovery and understanding of information about relevant startups in their first 18 months. So, we aim at providing a unified knowledge graph where:

1. Entities are uniquely identified by URIs and interlinked across sources.
2. Such entities are relevant to HAVAS 18 Labs, including startups, people of interest related to them through different roles, e.g. founder, investor, etc., bigger and more established companies, universities, and technology trends.
3. Rich information is provided about entities (facts, relationships and features).

To this purpose, the graph follows a lifecycle very similar to the one described in [5], comprising three main phases: knowledge acquisition, integration, and consumption. We extract data from online sources, including generalist and specialized web sites, online news, entrepreneurial and general purpose social networks, and other content providers. By maximizing the use of sources offering web APIs, we expect to minimize additional unstructured data processing time and complexity, at the cost of unexpected changes in the APIs, a potential source of decay in the knowledge graph. Data sources include:

- **Core data** from specialized sites like AngelList and CrunchBase, with useful facts about the main entities in the graph (startups, innovators, investors, other companies and universities), the relationships between them, and domain-specific news.
- **Relationships**: Beyond factual knowledge about the entities, the resulting graph makes emphasis on how they are related to each other. We enrich the relationship graph with information from Facebook, LinkedIn, and Twitter,
which helps completing a social and professional graph between the entities. Such explicit relationships support the discovery of new insights and navigation.

- **Extended media coverage**, with general news coming from media in any domain through Newsfeed.ijs.si. News text is processed with iSOCO’s semantic annotation component KTagger [1] in order to resolve entities and disambiguate.

We structure and integrate these data in an RDF dataset. The underlying schema is built on top of a number of standard W3C vocabularies, including Schema.org, FOAF, and SKOS, and IPTC’s rNews. On top of the data, a service layer is provided through a RESTful API with JSON and API key authentication. The API allows the exploitation of the graph by application developers and ultimately media business strategists through analytics platforms and dedicated user interfaces. API services include CRUD methods for entity and relationship management, graph navigation, search, and definition and access to business KPIs about the startups.

In addition to automated information extraction means, the knowledge graph can also be populated with on-site information by local rapporteurs, members of the local entrepreneurial scene distributed at each of the HAVAS 18 Innovation Labs. Rapporteurs are provided with means to add or modify entities and relationships in the graph, following the schema, assisted by autocomplete functionalities that leverage the knowledge previously stored in the graph. They also play the role of curators of knowledge produced either by other peer rapporteurs or extracted automatically. The combination of automatic methods and human expertise allow that a common knowledge graph can be leveraged consistently across the company.

Currently, the graph contains information about 1.812 startups, 559 technology trends, 1.597 innovators, 20 companies and 35 universities and research centers in Siliwood, following the Linked Data principles. All these entities are additionally connected to relevant online news, where they are mentioned (currently, 36,802), for extended and up-to-date information about them. The Knowledge Graph is updated daily in an automated batch process, identifying new entities and updating existing ones. We expect the knowledge graph to quickly reach the threshold of 300,000 startups below 18 months and extend to the remaining Labs in the next few months.

### 3 Value proposition

Innovation is often misunderstood and difficult to integrate into companies’ mindset and culture. So, why not activate relevant external talent and resources when necessary? The discovery and surveillance of trends and talent in the start-up ecosystem can be time consuming, though. Our knowledge graph sets its semantic engineering to run a surveillance monitoring of the entrepreneurial digital footprint, collecting and gathering fruitful insight and information, which provides our Innovation Labs staff with clear leads for their analyses. This is the key approach and philosophy of HAVAS. By automating part of the research process, we can get there faster and more accurately than competitors, leveraging millions of data points, and implementing consistency through a single and shared knowledge entry point.

At the moment this assisted process is integrated with a manual audit of trends and start-ups, executing a series of evaluation matrices to weigh and assess each individual entity in the graph against HAVAS’ business needs. The knowledge graph
is being opened to HAVAS’ network, with teams in 120 offices around the world and clients, providing access to knowledge about best-in-class talent to implement new thinking and cutting edge solutions to the never ending and evolving challenges within the media industry. Based on the knowledge graph, teams also rate and share experiences, ensuring that learning can be propagated across the network.

4 Discussion and future work

To optimize the trustworthiness and accuracy of the graph, we maximize the use of authoritative and specialized sources and prioritize freshness over volume. However, entity resolution and disambiguation is an issue, especially when unstructured data from unbounded domains come into play. During data integration and enrichment, several candidate entities can be identified. In order to resolve the correct one we define evidence models on top of the schema with the key classes required to provide an entity class with univocal context information. For example, in the case of Startup, this could be Founder, Client, and Technology. Upon data harvesting, e.g. from news, the text is processed by KTagger, which extracts entities and matches them against the evidence model providing a measure of evidence based on the context fragments identified in the text and allowing ranking. Further complexity is added when an entity which is not in the domain of interest, e.g. Domo, the gas station, has to be discriminated from one which is part of such domain and potentially in the graph, e.g. Domo, the startup. Other challenges include (sub)graph time and version management, reconciliation of automatic vs. human updates, and resilience against changes in the data sources, especially web APIs. Thus, it is key to monitor potential decay in the graph, through the application of existing techniques and methods for decay management [2,3]. Once structured as self-contained information packs, personalized subscription, delivery and recommendation of portions of the graph will also be possible. Finally, we will explore the possibility to release and interlink (parts of) the HAVAS 18 Labs Knowledge Graph to the LOD cloud for research purposes.

References

5. Torzec N. Yahoo’s knowledge graph. [Link](http://semtechbizsj2014.semanticweb.com/sessionPop.cfm?confid=82&proposalid=6452)
Deploying National Ontology Services: From ONKI to Finto

Osma Suominen\textsuperscript{1}, Sini Pessala\textsuperscript{1}, Jouni Tuominen\textsuperscript{2}, Mikko Lappalainen\textsuperscript{1}, Susanna Nykyri\textsuperscript{1}, Henri Ylikotila\textsuperscript{1}, Matias Frosterus\textsuperscript{1,2}, Eero Hyvönen\textsuperscript{2}

\textsuperscript{1}The National Library of Finland
\textsuperscript{2}Semantic Computing Research Group (SeCo), Aalto University, Finland

Abstract. The Finnish Ontology Library Service ONKI was published as a living laboratory prototype for public use in 2008. Its idea is to support content indexers and ontology developers via a browser interface and machine APIs. ONKI has been well-accepted, but being a prototype maintained by the ending research project FinnONTO (2003–2012), a more sustainable service was needed, supported by permanent governmental funding. To achieve this, ONKI was deployed and is being further developed by the National Library of Finland into a new national vocabulary service Finto. We discuss challenges in the deployment of ONKI into Finto and lessons learned during the transition process.

The Vision: A National Ontology Service

In Finland, a major research initiative FinnONTO \cite{4} was carried out in 2003–2012 with the goal of providing a national level semantic web ontology infrastructure based on centralized ontology services. Since 2008, a prototype of such a system, the ONKI Ontology Service\textsuperscript{1} \cite{9,8} has been used in a living laboratory experiment with more than 400 daily human visitors and over 400 registered domains using its web services, including the ONKI mash-up widget for annotating content in legacy systems and semantic query expansion. The FinnONTO infrastructure also includes the notion of creating and maintaining a holistic Linked Open Ontology Cloud KOKO that covers different domains, is maintained in a distributed fashion by expert groups in different domains, and is provided as a national centralized service.

In 2013, the Ministry of Education and Culture and the Ministry of Finance decided to finance the deployment of ONKI and its key ontologies into a sustainable, free, national service Finto\textsuperscript{2} (Finnish Thesaurus and Ontology Service), created and maintained by the National Library of Finland. Finto was opened to the public in January 2014, and the API services of ONKI were redirected to Finto in June. This paper summarizes, from a technical standpoint, the major ideas and components underlying Finto and lessons learned during the deployment process. Issues encountered in ontology engineering regarding, e.g., concept analysis and linguistic aspects have been discussed in a separate paper \cite{5}.

\textsuperscript{1}http://onki.fi
\textsuperscript{2}http://finto.fi
Deploying Ontology Services

ONKI/Finto supports the publication of ontologies by providing a centralized place for finding, accessing, and utilizing ontologies—the key functions of ontology libraries [2]. Using ontologies and integrating them in applications is made easier because different ontologies can be accessed via the same user interfaces and APIs.

User groups ONKI/Finto ontologies are used by both humans and machines. We identified three main human user groups, with slightly different needs in the user interface: full-time annotators (indexers working at, e.g., libraries and museums), users performing annotation as part of their other duties (e.g., journalists publishing articles), and ontology developers. In addition, users looking up information in databases use the service either directly, to look up suitable keywords, or indirectly via APIs used by the search system. The Finto user interface is designed to serve them all. For machine use and application developers, the service provides a variety of APIs with supporting documentation.

Finto service utilizing Skosmos software The development of a successor system for ONKI started within the FinnONTO project. The first step was ONKI Light [6], a prototype for an ontology browser on top of a SPARQL endpoint. The software has since evolved into Skosmos3, a vocabulary browser using SKOS and SPARQL, developed at the National Library. Skosmos provides a multilingual user interface for browsing and searching, and for visualizing concept hierarchies. The user interface has undergone repeated usability tests. The Finto service is set up as a specific installation of Skosmos, but Skosmos can be used to provide ontology services anywhere. As of September 2014, Finto serves more than 600 human visitors per day, of which 200 are returning users.

Skosmos relies on a SPARQL endpoint (Apache Jena Fuseki with the jenatext index) as its back end and is implemented mainly in PHP. The main benefits of using a SPARQL endpoint is that the data provided by the service is always up to date. This allows fast update cycles in vocabulary development. Vocabularies are pre-processed using Skosify [7] to ensure that they are valid SKOS. The source code is available under the MIT license.

Machine access to concepts The ONKI system offers machine access to ontologies not only by publishing Linked Data, but also custom APIs more suited for integration to e.g. document and collection management systems. ONKI provides three main APIs: a SOAP API, an HTTP API, and a JavaScript widget [8]. These have been integrated into systems used in museums, archives and libraries. For Finto and Skosmos, a new native REST API4 providing RDF/XML, Turtle or JSON-LD serializations was developed and API wrapper code implemented to support the ONKI APIs. The ONKI system is still available for browsing ontologies, but API calls to ONKI were redirected to Finto in June 2014. The Finto API serves more than 100,000 accesses on a busy day, while the old APIs receive between 10,000 and 15,000 hits.

3 https://github.com/NatLibFi/Skosmos
4 http://api.finto.fi
Ontologies in ONKI and Finto

At the heart of ONKI/Finto lies the General Finnish Ontology YSO. The National Library took over the development of YSO in 2013. Since then, special attention has been paid to making YSO intuitive and user-friendly without losing the benefits of machine-readable semantics. The top-level ontology has been reworked, multilingual aspects have been refined, and work is under way to link YSO to the Library of Congress Subject Headings (LCSH). The original OWL representation was changed to SKOS, with some extensions from ISO 25964\(^5\).

During the FinnONTO project, many YSO-based domain ontologies were created in collaboration with expert organizations. YSO is used as the central hub relating all the domain ontologies to one another while minimizing the number of direct links between them. The aim here is to facilitate the distributed development of the domain ontologies in expert organizations while allowing the domain ontology developers to worry only about links and changes to one other ontology. Their content is aggregated into the unified ontology cloud KOKO [3].

The work on this cloud of interlinked ontologies was begun during FinnONTO and has now been continued at the National Library. The ultimate goal is to use KOKO to relate the annotations in the datasets of the various organizations, facilitating interoperability and breaking down silos. KOKO has been in pilot use as an annotation vocabulary in, e.g., various museums and at the National Broadcasting Company YLE – in organizations that potentially manage material from all possible domains.

In addition to YSO, KOKO and the YSO-based domain ontologies, a number of thesauri, classifications and other controlled vocabularies have been published in Finto, including Medical Subject Headings\(^6\), Iconclass\(^7\), and Lexvo\(^8\) language codes, so they can be used via the same user interface and APIs. Currently 27 vocabularies are available and more are being prepared for publishing.

Experiences during Deployment

The process of transitioning users from ONKI to Finto has generally been smooth. The ONKI name had been used for several related but distinct initiatives, so the name Finto was chosen to avoid confusion. The Finto user interface has undergone multiple rounds of usability testing and is already well-liked by users, but the implementation could be further improved in terms of speed and scalability. The bottleneck is often the SPARQL endpoint.

During usability testing, the System Usability Score (SUS) [1] was calculated for the Finto system. The average score from 8 user tests was 80 points (out of 100), with individual scores ranging from 68 to 88 points. In earlier usability tests on the ONKI system, the average score from 14 user tests was 48 points.

\(^5\) http://www.niso.org/schemas/iso25964/
\(^6\) http://www.nlm.nih.gov/mesh/
\(^7\) http://iconclass.org
\(^8\) http://lexvo.org
The transition from ONKI APIs to Finto was managed by testing the new wrapper implementations both internally and among major user organizations well in advance of their deployment. Nevertheless, some problems had to be resolved after the transition, e.g., caching and SSL/TLS issues. Users were simultaneously introduced to new versions of ontologies including YSO and KOKO, which caused compatibility issues. However, those issues were generally resolved in a matter of days. We expect to encounter similar migration issues as ontologies evolve, but the changes are likely to be incremental in nature.

It remains a challenge to convert existing systems, processes and users to the ontology-based environment. Many databases that use thesauri in a traditional fashion (i.e., store terms only) are still in use, both within the library sector and beyond. There are generally few resources to convert legacy systems to make use of ontologies, URIs and APIs. The most significant competitor to Finto remains the VESA online thesaurus search service\(^9\), which is well-liked by indexers despite being 15 years old. In the future, we expect major growth in the usage of the Finto service, as well as more systems integrated to use the API services. We also expect organizations to deploy their own Skosmos instances to set up their own vocabulary services.

References


\(^9\) [http://vesa.lib.helsinki.fi](http://vesa.lib.helsinki.fi)
ePlanning: an Ontology-based System for Building Individualized Education Plans for Students with Special Educational Needs

Sofia Cramerotti¹, Marco Buccio¹, Giampiero Vaschetto¹, Marco Rospocher², Luciano Serafini², Elena Cardillo³, Ivan Donadello²,⁴

¹Edizioni Centro Studi Erickson S.p.A. - Via del Pioppeto 24, 38121 Trento, Italy
²Fondazione Bruno Kessler - Via Sommarive 18, 38123 Trento, Italy
³Institute for Informatics and Telematics—IIT-CNR, Via P. Bucci 17B, Rende, I-87036, Italy
⁴DISI, University of Trento, Via Sommarive 9, Trento, I-38123, Italy

The Individualized Education Plan (IEP) is a document that defines academic/life goals for a pupil with special educational needs. The IEP is built specifically for the pupil, and it is the result of a collaborative activity that involves the school special education team, the parents, other relevant educational stakeholders, as well as, whenever possible, the student. In details, an IEP specifies the student academic/life goals and the methods/kind of educational intervention to obtain these goals (long, medium, short term range). The IEP also identifies activities, supports and services that students need for being successful in their school activities in the perspective of the “special normality” principles, as required by the Italian Law 104/92 for students certificated for a disability. Besides the wide employment, in the last years, of IEPs in several Italian schools of any educational level (kindergarten, primary school, middle school, high school), the development of a IEP for a given pupil is a manual, complex and time-consuming activity.

To support and facilitate the building of the IEP, we developed a web-based decision support system, called ePlanning: users input relevant aspects of the profile of a pupil (e.g., age, diagnosis, observations) into the system, and based on this content the system guides the users in defining the more appropriate academic/life goals for the pupil, suggesting also activities and educational material that may help in achieving these goals. Semantic Web (SW) technology plays a key role in ePlanning, as well as in its development.

First, ePlanning (see Fig.1) is an ontology-based application, i.e., the content the system uses to support the construction of an IEP is encoded in an OWL 2 ontology, which formalizes: (i) processes, that represent functional abilities; (ii) relevant features of pupil profiles (e.g., age, school grade, a diagnosis in terms of a ICD¹ or ICF² code) and their relation with functional abilities; (iii) proposal of goals that can be set in the presence of an impairment of some functional abilities; and, (iv) activities and educational materials that

¹ http://www.who.int/classifications/icd/en/
² http://www.who.int/classifications/icf/en/
can be used to achieve the proposed goals. The system iteratively accesses the ontology content via querying, also exploiting inferred content materialized via OWL-DL reasoning.

An important example of query is the one that returns all the information of a given functional ability. Given the Uniform Resource Identifier (URI) of a functional ability the system connects to the data store containing the ontology, it performs the query with SPARQL language and retrieves all the relevant information. Such information are the parent (the URI of) and the children of that functional ability (according to the taxonomy of processes and sub-processes), its label, description and clarifying questions in natural language, the sex compatible to that functional ability, some possible ICF or ICD10 codes, an order and a weight representing its relevance in the taxonomy. Another query is the one that, given the URI of a process representing a functional ability, returns the information of its sub-processes. First the query retrieves all the sub-processes, and then the query above is executed for extracting the information of every single functional ability. The power of the semantic technologies is that the URIs of the individuals in the ontology univocally identifies them, so potentially (if the ontology would be public) a single functional ability could be retrieved by whatever application in the world.
The architecture of the ePlanning system is divided into three tiers: the Presentation Tier, the Business Logic Tier and the Data Tier. The Presentation Tier is the interface the user interacts with for building the IEP. It is the application oriented layer and it communicates its requests to the Business Logic Tier. The requests are handled by this latter layer through methods exposed by a web service implemented with a REST (REpresentational State Transfer) architecture. Every method semantically queries the ontology from the Data Tier in order to satisfy the application logic. The Data Tier physically retrieves the data from the ontology with the logical inferences already computed. The ontology is stored in an openRDF Sesame triple store.

Second, to favour the construction of a high-quality ontology, a heterogeneous team of (~20) users having complementary competencies and skill was involved in its development [1]:

- **Psychologists and Educators**: to define the taxonomy of processes and sub-processes (more than 400) referring to different functioning areas of the students: Cognitive – neuropsychological; Communication – language; Affective – relational; Motor skills; Sensory; Autonomy (personal and social); Learning.
- **Teachers** (from: kindergarten, primary school, middle school, high school): to define goals (long, medium, short term range) and related activities established on the base of the level of impairment (more than 9000).
- **Knowledge Engineers**: to provide the modelling expertise to properly model the rich content to be represented.
- **Application Engineers**: to bring in the application perspective, in particular for the requirements of application-specific content to be modelled in the ontology.

The modelling was performed with a customized version of MoKI [2], the Modelling Wiki, that was extensively used by the modellers: in over a one-year modelling period, we tracked more than 6500 editing operations from 13 distinct users.

ePlanning will be released in September 2014 as a commercial tool\(^4\) edited by Edizioni Centro Studi Erickson,\(^5\) having as target audience the schools of all the national territory.

In this talk we will discuss the experience of adopting Semantic Web technology in a key product of the enterprise, including a report of the lessons learned (i) in collaboratively building an ontology (a first experience for the enterprise and most of the users involved in

\(^3\) [http://moki.fbk.eu](http://moki.fbk.eu)

\(^4\) commercially released as “SOFIA”

\(^5\) an Italian Small-Medium Enterprise in the Publishing Domain
the modelling activities) in a concrete and multidisciplinary context, as well as (ii) in developing an ontology-based decision support system.

In particular, regarding point (i), we will report, for example, the importance of having a “flexible”, ad-hoc, on-line, and collaborative modelling tool as MoKI, that allowed for avoiding the proliferation of “latest” versions of documents by domain experts, familiar with spreadsheet before this experience, and consequently considerably reducing human effort. Furthermore, other two important aspects emerged during the modelling phase: the importance of early deploying the application ontology in its corresponding system (even if under development) already during the modelling activities, in order to favour the improvement of the ontology quality and the early detection of modelling mistakes and assumptions; and finally, the importance of adopting an hybrid ontological representation (i.e. representing each core element both as a class and as an individual) to ensure a multipurpose ontology, to be used as a traditional classification ontology on the one hand, and as the main data component of an application system on the other hand.

Regarding point (ii) we will report the importance of hiding the difficulties of retrieving semantic data from data store (e.g., handling URIs and implementing SPARQL queries) by exposing pre-canned methods through web service. The web service was implemented by the knowledge engineers, while the application engineers concentrated only on the application perspective without any efforts of interfacing with semantic data and without altering their usual development processes. This work methodology has allowed for a rapid development of ePlanning system.

These and other lessons learned may be beneficial for similar modelling initiatives regarding the development of ontology-based application in practical case.

Acknowledgements This work was partially supported by “ePlanning - Sistema esperto per la creazione di progetti educativo-didattici per alunni con Bisogni Educativi Speciali”, funded by the Operational Programme “Fondo Europeo di Sviluppo Regionale (FESR) 2007-2013” of the Province of Trento, Italy.

References

Ontology-Based Linking of Social, Open, and Enterprise Data for Business Intelligence

Tope Omitola¹, John Davies², Alistair Duke², Hugh Glaser³, and Nigel Shadbolt¹

¹ Electronics and Computer Science
University of Southampton, UK
{t.omitola, nrs}@ecs.soton.ac.uk
² British Telecommunications, UK
{john.nj.davies, alistair.duke}@bt.com
³ Seme4 Ltd., UK
hugh.glaser@seme4.com

1 Introduction

We are at the cusp of two major revolutions impacting the world of work and commerce. These are the rise of social media and the rise of Big Data. Data is everywhere. Each phone call, email, chat request, or person-to-person interaction between a customer and a brand provides organisations with invaluable information. This wealth of data can yield valuable information, such as revealing precious insight into customers’ needs and desires, allowing companies to personalise their services accordingly. A business revolves around its customers and their social connections. These social connections are valuable data that can be useful for enterprises.

For a company to thrive in this new world, these data need to be used to identify opportunities in new sectors, support employees, customers, and other external partners. These data can also be used to create a more intelligent understanding of customers, and to help predict future customer behaviour. This paper (and talk) will describe how we apply Linked Data to enable BT, in particular BT Business (BTB), a division of BT, to take advantage of this Data revolution. We use Linked Data technology to integrate internal and external datasets, including structured, unstructured, and social data. We describe (and shall expand on these in the talk) how this integration allows new and insightful information to be derived.

BTB, the UK’s leading provider of business communications services with over one million small and medium sized enterprise customers, would especially like to use Linked Data to solve these business challenges (amongst others):

- To manage and extract value from its disparate, isolated data,
- To take advantage of information from external, non-enterprise, and other social data in order to provide new, exciting, and useful services that create value for customers,
To identify trends and issues that are specific to circumstances such as competitor activity, products offered, industrial sector of the customer, and profiles of members of sales teams.

2 System Operation

Ontology-based data access and management (OBDM) is a methodology that is used to access, integrate, and manage data in big enterprises. It consists of a three-level architecture constituting an ontology, the data sources, and the mapping between the two. We have applied the ideas of OBDM for our data integration process. Our systems consists of a Unified Ontology, and using this ontology to guide us to transform the datasets of four systems, into Linked Data.

a. The Unified Ontology: This consists of entities such as the concepts of a BT employee (BTEmployee), of a BT employee (EmployeeSocialMediaUser) that is also a member of a social network, of sales team members (SalesForceUser), of client companies (Account), and industrial sectors (SICCode). (We will expand on this in the talk).

b. Four different systems are involved in the process: (1) Public information of clients of BTB as derived from OpenCorporates; (2) Members’ social connections as derived from LinedIn; (3) an LDAP-backed General Employee Data store; and the BTB Win-Loss system (which stores a collection of CSV files of companies that clients of BTB). We transform most of the data in these systems into RDF, linking them together using appropriate class and data instances’ URIs.

Data Consumption The linked data platform that is constructed allows us to make very flexible Sparql queries, which we shall describe further in our talk at the Workshop.

3 Conclusions

This paper described how we linked social, open, and enterprise data for Business Intelligence, and how this has been applied in a telecommunications company. Some of the challenges we found include (a) the establishment of a strong business case and the availability of a ‘data champion’ to help bring the stakeholders together; (b) Data Discovery and Provenance. Discovering the appropriate datasets with the right provenance criteria can be time-consuming but very important; (c) Data Cleaning and Interlinking. Many of the discovered datasets may not be in the appropriate format to be useful, so they need to be “cleaned”. The choice of URIs to use for interlinking datasets is application-specific. This choice should be guided by the business case; and (e) Data Modelling. A goal of integration is to have a unified view of the data from the disparate sources. Having a unified ontology helps towards this. Future data integration tasks need to think of these challenges and provide the appropriate solutions. We shall provide more details of these challenges in the talk.
4 Acknowledgments

This work is supported under SOCIAM: The Theory and Practice of Social Machines. The SOCIAM Project is funded by the UK Engineering and Physical Sciences Research Council (EPSRC) under grant number EP/J017728/1 and comprises the Universities of Southampton, Oxford, and Edinburgh.
1 Introduction

The BBC has understood the value of online learning from the early stages of the web, and has provided rich educational material to those wanting to learn. An example of this is the BBC Bitesize website\(^1\), which started back in 1998 and is a popular formal education resource.

In the formal learning space the BBC has a number of sites: the already mentioned Bitesize, Skillswise\(^2\) and Class Clips\(^3\) amongst others. There are tens of thousands of content items across these sites, with each site having different mechanisms for publishing, discovering and describing the content it serves.

Moving away from a disconnected production process, a model for describing content consistently in the context of the UK national curricula was developed. This model provided the foundation for building the new Knowledge & Learning beta website, presenting learning resources in the context of the UK national curricula in a consistent way. In addition, it allows for consistent reflection of changes in the national curricula throughout the product.

Designing the architecture of such a system is a challenging task. Each of the existing sites have similar yet different ways of describing and navigating through their content. In addition, the existing learning sites do not have a single content description model that could be reused in the beta site. Having a flexible structure in the back-end that can reflect the national curriculum and that can be used for consistently describing and organising learning resources is a key feature of the architecture.

We are going to present the architecture behind the Knowledge & Learning Beta site and we focus on the curriculum ontology, which is central to the architecture. We show how it is used to describe and organise learning resources, how it supports navigation and how it is aligned with semantic markup vocabularies for better precision in search. We will also present some of the challenges we faced and discuss future work.

---

1. [http://www.bbc.co.uk/bitesize/](http://www.bbc.co.uk/bitesize/)
2. [http://www.bbc.co.uk/skillswise](http://www.bbc.co.uk/skillswise)
3. [http://www.bbc.co.uk/learningzone/clips/](http://www.bbc.co.uk/learningzone/clips/)
2 Architecture Overview

The overview of the architecture of the architecture is shown in Figure 1. The curriculum ontology plays a key role in the description of the learning resources, which are video clips and revision guides.

Following a Dynamic Semantic Publishing (DSP) model, the BBC Knowledge & Learning system incorporates semantic web models and linked data in its architecture. In the Knowledge & Learning Online Pages, different components of the front end are served by different systems in the back-end.

In a nutshell, the ontology and its instance data are served by the Linked Data Platform, which amongst other services it provides the BBC internal triple store. Learning content such as video clips and revision guides are saved in a different system named as Content Store. The coupling between Learning resources and curriculum ontology is done through semantic annotation. In particular, the content items are tagged with curriculum instances so that the Application Layer can retrieve related content for curriculum ontology instances.

The Curriculum Ontology aims (1) to provide a model of the national curricula across the UK, (2) to organise learning resources, e.g. video clips and revision guides and (3) to allow users to discover content via the national curricula. These are achieved by providing broad units of learning (e.g. a Topic) and more finely grained units (e.g. a Topic of Study). The core curriculum data model is shown in Figure 2. Level refers to different stages of education like Key Stage ‘KS3’, ‘GCSE’ etc. Fields of study like ‘Science’ and ‘Maths’ are high level disciplines of the curriculum. Programmes of study like ‘GCSE Maths’, ‘KS1 Computing’ are expressed with respect to a nation, level and a field of study. Topics of study are topics within the context of a programme of study. An example is the ‘Digital Literacy’ topic of study of the ‘KS1 Computing’ programme.

The Curriculum Ontology is the glue that holds the content together and the basis of the website navigation. One of the benefits of this is that it can offer dynamic aggregations of content achieved by querying the linked curriculum
data. It can also help to easily discover content. For example, the recommendations on other relevant topics to a video clip is done via the ontology data. Building additional recommendation services using the curriculum ontology and other BBC and external ontologies is a promising direction.

Mapping the Curriculum Ontology concepts with learning markup vocabularies, such as the Learning Resource Metadata Initiative (LRMI)\(^4\), can allow for better precision in search using the metadata of the learning content. Aligning the curriculum ontology with the LRMI vocabulary and enriching the BBC education online pages with the LRMI standard markup can provide future benefits as search engines refine their results on such markups.

The integration of semantic web technologies in the architecture of the BBC education online pages was a key decision when moved to a consistent way of describing content. Keeping a controlled vocabulary as linked data and having a consistent workflow for data management can improve the quality of the data and highlights potential areas of duplication. It also allows for publishing the data under a permissive license and integrating them with external resources (e.g. DBpedia\(^5\), Freebasehttps://www.freebase.com/ etc). There is also potential in discovering and promoting across BBC different types of content, which has been semantically annotated with linked data.

Hiding the ontology’s complexity and providing a consistent navigation is a challenge of the architecture, which is achieved with the implementation of services and APIs on the top of the linked data. Tooling and data management interfaces are also important and it is currently an area area that is being concentrated on.

\(^4\) [http://www.lrmi.net/the-specification](http://www.lrmi.net/the-specification)

\(^5\) [http://dbpedia.org/](http://dbpedia.org/)
Abstract. The SICRaS project aims at deploying semantic, big data and geospatial technologies in the industry for the purposes of tax assessment and social policy making. We believe that the combination of these technologies can enable the efficient reasoning on complex scenarios.

Keywords: Fiscal policy making, Semantic Technologies, Entity Recognition, Geospatial Technologies

1 Introduction

In these years, Italy is dealing with serious economic and social issues, which are aggravated by the recent global economic crisis. These issues include, among others: high fiscal burden, widespread tax evasion, increasing unemployment rate and the progressively aging of population. In this context there are two major needs that public administrations have to meet. On one hand, it is mandatory to fight tax evasion, on the other hand, there is the need to ensure social services in a more efficient, fair and effective way.

Our industrial need, which we are developing within the SICRaS project, is to support governance and policy making in achieving these goals through the integrated analysis of a large amount of information collected by both public administrations and other public officers and organizations (e.g. notaries, public utilities and so on). This information is typically scattered over several heterogeneous and decoupled data sources. Moreover, it might also be partially outdated, unreliable and redundant.

The adoption of semantic technologies enables the construction of an integrated, trustworthy and accurate knowledge base, providing a picture of the

* This work has been partially funded by the Autonomous Province of Trento (Legge 6/1999, DD n. 251) and by the European Commission (Grant No. 296448)
fiscal and social situation of each single citizen and of the community of a local municipality, overcoming the limitations of legacy systems. Cornerstones of the solution are: 1) a set of domain ontologies aimed at improving data integration, and at producing useful inference from explicit information; 2) a scalable system to reconcile identities to the same real world entity across datasets, associating a unique and persistent name to each single entity; and 3) leveraging geo-spatial technologies to achieve a deeper understanding of the observed districts by means of spatial analysis and reasoning.

2 Towards a Linked Data for tax domain

Tax information systems typically work with an extraordinary amount of data concerning many different aspects of taxpayers life: personal and company details, cadastral information, job positions and so on. The role of ontologies in data exchange and integration so as to retrace tax positions from all these information is definitely invaluable [1]. In fact, since data are collected at various times by different partner administrations, there is the need to link all these tax related information from distributed source streams. Looking at the domain, we recognize that tributes are generally imposed taking into account of specific circumstances or events that happen during the taxpayers life: the taking up of residency in a new town, join a nuclear family, buying a house, etc.

Given these assumptions, we chose to follow an entity/event based ontological modeling approach. The goal is to support an integration pipeline, producing a unified view of the relevant fiscal facts scattered in datasets supplied by diverse public institutions. Our modeling approach allows then to materialize, at a given instant of time, the deduced tax position involving each single taxpayer.

3 From (Big) Data to (Big) Knowledge

One of the known limits of semantic technologies is scalability both in reasoning and data management. This has twofold justification, on the one hand there are limits related to the computational complexity of reasoning based on model theoretic semantics, on the other hand the immaturity of existing technologies for data management. To overcome these limits, we organized a pipeline relying on an ensemble of scalable and state of the art technologies to define a Semantic ETL suitable to create a Semantic Big Data Pool.

We rely on a customized and optimized version of Open Refine tool to perform data cleaning operation, including syntactical validations and transformation of the original data coming from the public institutions. The formal and syntactical validations, expressed according to a specific rule language, are the results of many years of experience on the field. At this stage, issues related to semantic and structural heterogeneity affecting the original data are normalized relying on a set of maintainable contextual ontology mappings towards the defined domain ontology. Each record is analysed to extract information about the involved entities to reconcile their identities relying on the Okkaum Entity Name System [2].
Once the identity of any entity involved in each of the records has been disambiguated, the dataset is exported in RDF and stored as many entity-centric named graphs into the Hadoop Distributed File System (HDFS). The result of this first part of the process is a physically distributed and logically integrated large RDF graph (described in Table 1) that can be manipulated and processed relying on emerging big data technology. Therefore, we rely on tools like Apache HBase, Apache Incubated Spark, Apache Incubated Flink, Apache Hive, and Apache Pig to define (complex) big data shuffling processes producing any view, analysis and mesh-up necessary to support tax assessment domain applications. In fact, it is possible to select subsets of the giant RDF graph to store it in application specific data management systems. For example, we sink data into a triple store such as OpenRDF Sesame and enable scalable rule-based reasoning tasks using SPRINGLES. Another example is to build sub-graphs to support seamless real time navigation of the knowledge relying on effective indexing tools such as Apache SIREn, or to perform graph-based analysis sinking data in a graph database (e.g. Neo Technology Neo4J).

### 4 Geographic technologies for spatial analysis and reasoning

The integration of geo-spatial technologies adds an important analytic dimension to SICRaS. Taking advantage of this kind of information, we firstly intend to exploit the notion of territory, seen as a spatial region in our ontological model. This enables new ways of extracting, observing and analysing data about real world entities and the spatial relations among them. Secondly, we develop techniques to match entities relying on geo-spatial features to link our knowledge base to external sources (e.g. urban development plans) to find out new information valuable for tax assessment and for other fiscal and social purposes.

### 5 Big Analytics

Analytic functions will be developed on top of the SICRaS Big Data. To do so, we are integrating semantic technologies in the core of the SpagoBI suite [3]. Big Analysis techniques will then be employed for facilitating tax and social processes with advanced intelligence functions such as, for instance:

<table>
<thead>
<tr>
<th>Volume</th>
<th>Velocity</th>
<th>Variety</th>
</tr>
</thead>
<tbody>
<tr>
<td>(O(10^{12}))</td>
<td>(O(10^{16}))</td>
<td>24 different structured types and any semi-structured or destructured Web Resource</td>
</tr>
</tbody>
</table>

Table 1. Characteristics of SICRaS Big Data.
– **Identifying behavioral patterns:** identifying common traits of either recognized tax evaders or individuals which suffer the same conditions of social exclusion’s risk;  
– **Identifying citizens exhibiting specific behavior or characteristics:** recognizing citizens that behave according a given tax evader profile or that might be exposed to the risk of social exclusion;  
– **Analyzing trends and predicting possible changes in the territory:** analyzing social or fiscal relevant phenomena to predict their evolution in time and over the territory. For instance, this technique could be employed to study how the number of nuclear families consisting of a single elderly person changes in time and in distribution over the territory.

Such decision support functions will also take advantage of Spatial Business Intelligence (SBI) realized by integrating Geo-Spatial and Business Intelligence technologies. The combination of traditional and spatial data will allow us to employ spatial analysis and map visualization to offer innovative and ground-breaking intelligence solutions for local policy makers.

6 Deployment perspectives and concluding remarks

In SICRaS we define a scalable and efficient data processing pipeline, capable of overcoming the limits of current semantic technologies riding the wave of emerging big data processing tools. A wise union of semantic and big data technologies, tempered with deep domain knowledge and sophisticated geospatial tools, creates seamless opportunities to define tax assessment solutions.

Exploiting the wealth of data in an efficient and effective way, we aim to define the next generation of tools for policy makers and help the Italian institutions in overcoming the challenges of the 21st century. In fact, by the end of 2015, we plan to test our solutions in cooperation with three big Italian municipalities which will be SICRaS’ early adopters. After this test phase we plan to consolidate the SICRaS product line and to begin a marketing operation starting from our current customers (more than 600 Italian municipalities). In the future, the SICRaS Big Knowledge will also enable new Smart Cities’ tools.

References

Using Semantic Technologies to Mine Customer Insights in Telecom Industry

Rajaraman Kanagasabai, Anitha Veeramani, Le Duy Ngan, Ghim-Eng Yap
Data Analytics Department,
Institute for Infocomm Research,
A*STAR Singapore
{kanagasa,vanitha,dnle,geyap}@i2r.a-star.edu.sg

James Decraene, Amy Shi-Nash
R&D Labs, Living Analytics,
Singapore Telecommunication Ltd, Singapore
{jdecreane,amyshinash}@singtel.com

Abstract. Many telecommunication companies today have actively started to transform the way they do business, going beyond communication infrastructure providers and repositioning themselves as data-driven service providers to create new revenue streams. In this paper, we present a novel industrial application where semantic technologies are successfully used to mine commercial interactions of mobile customers, to get new aggregated insights from their call records.

Keywords: Big data analytics, Ontologies, Semantic Classification, IAB taxonomy, Marketing & Advertising

1 Background

Many telecommunication companies (telcos) today have actively started to transform the way they do business, going beyond communication infrastructure providers and repositioning themselves as data-driven service providers to create new revenue streams. New business opportunities, notably in market research, can be realised using telco data especially when it is complemented with external open data sources. Indeed, significant efforts have gone into mining customer insights from the massive mobile network data while preserving the end customers’ privacy. In this paper, we present a novel industrial application where semantic technologies are successfully used to mine commercial interactions of anonymous mobile device users, to get new aggregated insights from their call records.

2 Our Method

The case study is inspired by an observed semantic gap between the contextual business categories that can be derived from the customers’ call records and the industry-standard classification scheme that is often a requirement for consistent ad
targeting. In particular, the *Interactive Advertising Bureau (IAB) Contextual Taxonomy* [1], with 23 Tier-1 classes and 371 Tier-2 classes, is an international standard that is adopted e.g. by the Google Display Network. Mapping from thousands of contextual business categories to such a concise taxonomy often involves one to many mappings and thus is a daunting task for the market researchers.

Traditional approaches using machine learning techniques like the Support Vector Machines (SVM) [2] can help automate this task but it is not straightforward when applied over thousands of categories spread across a wide variety of domain areas. Our experience mapping a total of 2532 contextual business categories to the IAB Taxonomy shows that applying text feature extraction and matching resulted in merely 263 matched categories (approximately 10% in recall). The challenge is that this is in fact a large-scale multi-class multi-label classification problem that needs sufficient training examples to generalize well. Obtaining such training labels specific to each application is expensive and it is not feasible to repeat this for all applications. In this paper, we leverage domain knowledge in semantic machine learning methodologies and avoid the need to invest in expensive training data.

Using public knowledge bases WordNet [3], DMOZ [4], and Yahoo! Answers [5] as our domain ontology model references, we investigated the three different semantic IAB classification methods described below:

I. We employ WordNet features to build an extended text vector for classification, and use it as a baseline for comparison.

II. DMOZ Open Directory is among the largest human-curated directories online. We ingested RDF dumps of DMOZ into AllegroGraph triplestore as our DMOZ ontology model. Low-level text features were extracted from each contextual category to find its semantically-matching categories in the DMOZ ontology and the DMOZ categories are used to find best IAB classes.

III. Yahoo! Answers is a community-driven Q&A site hosted by Yahoo! Inc. The site categorizes questions and answers in a shallow categorical hierarchy that is similar to IAB, though the category names are very different. We capitalize on this by first searching Yahoo! Answers with the contextual category and using the returned Yahoo! Categories to match IAB classes.

We built a corpus with just 525 contextual business categories and created IAB class assignments by using two human experts to classify independently and a third expert to cross-check the assignment. The corpus was used in our research to fine-tune parameters in all the three classification methods. Following that, we validated the methods on a full set of 2532 contextual categories and manually evaluated their classifications. Method III performed best, followed by Method II and then Method I.
3 Deployment

A customer insights dashboard product was developed to provide user behavioural insights based on users’ geo-location traces and call details records (“who called who”). All records were anonymised via a one-way AES encryption-hashing process and neither personal data nor calls content was used.

A key offering of the product is its market segmentation service where in-depth user profiling was conducted to infer various people traits of interest such as demographics, occupation, housing type, and travel pattern. To illustrate, the example screenshot in Figure 1 shows the distribution of work locations for people who are living within a specific location inside Singapore (marked as a blue cell).

![Figure 1: Example screenshot of the customer insights product showing work locations distribution and commercial interactions profile of people living in Buona Vista (blue cell) – the commercial interactions profile was successfully generated using the novel Yahoo! Answers-based semantic IAB classification method presented in this paper.](image)

To enrich the customer profiling, we used call detail records and extracted calls made to local businesses. This was conducted through identifying the “business numbers” being called and extracting the associated contextual business category from open data that was publicly available online. We applied Method III (using Yahoo! Answers) to map the arbitrarily-defined 2532 contextual business categories into the IAB Contextual Taxonomy. As shown in Figure 1, the proposed method enables the end product to effectively provide customer insights in terms of market segments based on such commercial interactions of mobile device users.
References


Abstract. The growing content in online media libraries results in an increasing challenge for editors to curate their information and make it visible to their users. As a result, the users are confronted with the great amount of content, unable to find the information they are really interested in. The Smart Media Navigator (SMN) aims to analyze a platform’s entire content, enrich it with Linked Data content, and present it to the user in an innovative user interface with the help of semantic web technologies. The SMN enables the user to dynamically explore and navigate media library content.

Keywords: recommender systems, interface design, dbpedia, linked data, semantic web

1 Introduction

Content in most of today’s online blogs and media libraries is arranged in chronological order, which means that the top entries receive the most attention and older yet relevant content remains hidden to the users. Linking current posts entries to older ones manually is one possibility, but contrariwise linking older content to current posts requires much time and effort and thus, cannot be achieved by most editors. In order to search and retrieve the content of multimedia platforms, manually edited tags are generally used, which lack completeness and often are ambiguous and heterogeneous. A standardized vocabulary of tags may limit the editors in creativity and topics of interest that appear in future entries may not be considered. In case the user is interested in further information on a specific topic, which is actually not part of the platform, she usually has to leave the platform and may never return.

The solution to overcome these problem are intelligent recommender systems which are also able to help the authors to structure their content in a way that navigation and retrieval is aligned with the users needs as optimal as possible. But, many recommender systems rely on usage data and suffer from a general cold-start problem, when they are applied to a new content library without an established user community. The most approaches incorporate natural language
processing and therefore have to cope with ambiguity. By making use of formal knowledgebases, e.g. provided as DBpedia\(^1\), semantic technologies help to improve the quality of recommender systems for language processing, recommendation generation, and content enrichment.

## 2 Smart Media Navigator

The **Smart Media Navigator (SMN)** is a navigation and recommendation framework, which uplifts a platform’s content by enriching it with Linked Open Data (LOD). Therby, the content is complemented with additional information from the underlying LOD knowledgebase DBpedia. To support the user in navigating and exploring the content, semantic relations between different content items are visualized in a browsing interface.

As an online recommender system based on Linked Open Data, the main target groups for the SMN are broadcasting companies with online media libraries, archives with multimedia content, video-on-demand platforms, or blogs. The SMN aims to improve the user’s and author’s experience while curating and navigating the content. The first release will be implemented as a plugin for the blogging platform Wordpress\(^2\). Four main features (see Fig. 1) will be implemented in the first release:

1. **Automated mapping**: Fig 1 depicts the wordpress editing interface, new buttons were created in tinyMCE\(^3\) in order to automatically annotate paragraphs with DBpedia entities. For this step, yovisto’s named entity mapping was integrated through a RESTful\(^4\) webservice. With the possibilities to automatically annotate the content and manually edit the annotations, editors and authors can decide, which entities they really want to have linked and enriched with the DBpedia data.

2. **Embedded RDFa**: The annotated information is directly embedded as RDFa in the articles HTML markup. When a webservice is triggered, this information is extracted out of the blog post with an Apache Jena\(^5\) based RDFa extractor implementation. The extracted entities are stored in a local triple store\(^6\).

3. **Enrichment** In order, to import additional information about the extracted entities, a federated SPARQL update\(^7\) query is executed incorporating the public SPARQL endpoint of DBpedia\(^8\). It imports all triples the entity is involved as

\(^1\) [http://dbpedia.org/](http://dbpedia.org/)
\(^2\) [https://wordpress.org/](https://wordpress.org/)
\(^3\) [http://www.tinymce.com/](http://www.tinymce.com/)
\(^4\) [https://jersey.java.net/](https://jersey.java.net/)
\(^5\) [https://jena.apache.org/](https://jena.apache.org/)
\(^6\) [http://jena.apache.org/documentation/tdb/](http://jena.apache.org/documentation/tdb/)
\(^7\) [http://www.w3.org/TR/sparql11-federated-query/](http://www.w3.org/TR/sparql11-federated-query/)
\(^8\) [http://dbpedia.org/sparql](http://dbpedia.org/sparql)
object or subject from DBpedia to the local triple store. This practice has the advantage, that for efficiency and availability reasons there is no local copy of DBpedia necessary. The update is only necessary once when the article, or when the DBpedia changes. Furthermore, the local triple store only stores the triples, which are really used. It stays as small as possible and therefore efficient.

The imported additional information is then displayed on the articles web page. If the user hovers over an annotated entity in the articles text, additional information, such as abstracts and depictions, is appearing in a tooltip (see Fig.2).

4. Visualization The imported entity information is used to find associations (direct connections and connections of path length two) between entities of different articles, and therefore enables to recommend blog posts related to a given entity. These associations are used to create the relation browser, as depicted in Fig. 3. A main entity from a blogpost is automatically chosen (e.g. Albert Einstein) and based on this entity, further related entities are visualized in the relation browser above. They are sorted by persons, places, other 'things' and possibly also events. Hovering one entity (e.g. Zurich) reveals its relations to further entities and the main entity. Clicking on the entity Zurich in the relation browser turns it into the new main entity and the relation browser re-arranges according to Zurich. Next to the navigation feature, recommendations are shown based on the entities, the user may be interested in.

3 Conclusion

In conclusion, editors can take advantage of an integrated annotation tool, which helps to map entities with DBpedia as they write and thus, helps to reduce their workload. The additional information retrieved from DBpedia in combination with the relation browser will not substitute but improve traditionally arranged online multimedia libraries without altering their actual content. It increases the existing content’s value and motivates the platform users to stay in the site.

The first showcase using the plugin will be the yovisto blog\(^9\), which currently contains more than 800 daily articles consisting of texts, images, and videos. A demonstration of SMN will be given at the conference. The SMN is a project supported by the MIZ Potsdam-Babelsberg\(^10\) and will be finished by March 2015.

\(^9\) http://blog.yovisto.com/
\(^10\) http://www.miz-babelsberg.de/
Metropolis — A Cinematic Vision of Technology and Fear

On January 10, 1927, German expressionist epic science-fiction film Metropolis was premiered in Berlin, being the first feature film of its time.

Metropolis is a 1927 German expressionist epic science-fiction film directed by Fritz Lang. The film was written by Lang and his wife Thea von Harbou and starred Brigitte Helm, Gustav Fröhlich, Alfred Abel and Rudolf Klein-Rogge. A silent film, it was produced in the Babelsberg Studios by UFA. It is regarded as a pioneer work of science fiction movies, being the first feature length movie of the genre.

Fig. 1. Automated Annotation

Fig. 2. Enrichment

Fig. 3. Relation Browser and Recommender
Abstract— In this paper, we present the application case of semantic IoT platform technology to WISE project in Korea Meteorological Services. Current M2M platform technology applied to weather service is mainly focused on remote data collection. Therefore, it is difficult to analyze the domain context for decision support and provide the better customized semantic related weather information. In WISE project, big data such as high-resolution weather data and model data are collected. Moreover, it aims to support the interoperability and convergence of IoT data for urban and rural meteorology services.

I. INTRODUCTION

With the development of new communication technology and sensor technology, various attempts for supporting human life more conveniently are increasing. M2M technology is focused on the remote sensing and transport information to another machine via cellular network. Weather information service based on M2M technology is limited to a simple data acquisition and processing and is insufficient to provide a data integration and interoperability. On the other hand, more recent research on IoT (Internet of Things) technology[1] is trying to allow things to judge and operate autonomously and collaboratively by working through the Internet between things.

Recently, the attempts for applying semantic web technology to decision support M2M/IoT services are increasing. E.g. the SemSorGrid4Env project of EC[2] and Australian Climate Observations Reference Network - Surface Air Temperature (ACORN-SAT) dataset[3]. Unlike conventional meteorological services, semantic weather information service combined semantic web technology should be able to solve the data interoperability of diverse weather data and application domain data. However, due to the diverse and heterogeneous nature of IoT data, it is difficult to analyze user’s context and better customized weather related semantic information. Current M2M platform technology applied to weather service is mainly focused on remote data collection. Therefore, it is difficult to analyze the domain context for decision support and provide the better customized semantic related weather information. As shown in Figure 1, in WISE project, big data such as high-resolution weather data and model data are collected. Moreover, it aims to support the interoperability and convergence of IoT data for urban and rural meteorology services.

In this paper, we present the application case of semantic IoT platform technology to WISE project in Korea Meteorological Services. More in detail, the platform architecture and developed semantic technology is described.

II. WISE PLATFORM

WISE[4], which is a recently launched project of the Korea Meteorological Administration (KMA) aimed at developing a next-generation Weather Information Service Engine (WISE). WISE represents an investment over eight years for efforts to resolve urban environmental issues, through scientific advances in high-resolution weather forecasting, urban flood prediction, road meteorology and urban carbon dynamics, and new urban service systems to minimize and mitigate the impacts of natural disasters and climate change on urban dwellers. The main objectives of WISE platform are the improvements of technology & infrastructure for the implementation of urban & rural meteorology information services, decision support for disaster relief, information production support for national agenda, and building up a mashup service platform for easy customized services.

As shown in Figure 2, WISE Platform consists of three subsystems such as M2M platform, Semantic IoT platform and user service platform. Using M2M platform, diverse source of high-resolution weather data should be collected remotely and...
stably. Realtime M2M data and legacy weather information are stored in cloud DB which provides scalability and high-performance. Big data from cloud DB can be translated into new semantic knowledge by integrating with domain data and LODs. Semantic IoT platform provide semantic annotation and semantic processing. The translated semantic data, which is RDF base data, is managed into semantic repository. Using the semantic open API of semantic IoT platform, various user portal services are supported by the user service platform.

III. WISE SEMANTIC IOT PLATFORM

The semantic IoT platform consists of five main modules as Figure 3: semantic ontology, semantic processor, semantic query engine, semantic repository and semantic open API.

A. WISE Platform Ontology

Several kinds of ontologies are defined to support the WISE semantic service: platform ontologies, service domain ontologies and service ontologies. Platform ontologies mean the commonly applied ontologies that are independent with specific WISE service. Figure 4 show the relationships of WISE ontologies.

Therefore, platform ontologies generate description and process information to derive the abstracted real world event from sensing.

<table>
<thead>
<tr>
<th>TABLE I PLATFORM ONTOLOGY INPUT/OUTPUT DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Type</strong></td>
</tr>
<tr>
<td>INPUT</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>OUTPUT</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

B. Semantic Processing

The semantic processor performs the sensing data translation by using the translation rules and WISE ontology model. The semantic translator is a processor for converting non-semantic data(non RDF data) to semantic data(RDF data). The translation rules define the method of mapping each elements of the RDF triple pattern into the target ontology model or the value and type of the literal.

The translated RDF data are stored into semantic repository. To support scalability and performance of inference, semantic repository is implemented by using Hbase of Hadoop platform. Due to the nature of distributed and parallel processing of Hbase, our repository shows more high performance than any other existing RDF repositories.

C. Semantic Queries and Open API

The platform provides semantic query interface of SPARQL. The WISE applications or user service platform can query to derive more abstracted knowledge from semantic repository. In order to use the semantic platform easily, semantic query browser /visualization tool are required as shown in Figure 5.

IV. IMPLEMENTATION AND CONCLUSION

In Korea, the solitary senior care system that monitors the status of the solitary senior efficiently is in operation. Recently, due to the abnormal weather condition, the damage of heat wave is increasing. The figure 6 shows the example of applying the WISE system to solitary senior care system. The micro-scaled weather information and the WBGT expectation information can be easily related with the other system such as the solitary senior care system as well as easily infer the semantic query result depending on the user type.
The implemented semantic IoT platform was applied to WISE project to generate semantic weather data and to provide better customized semantic related weather service. Now, in order to prove the more detail benefits of semantic IoT platform in the WISE project, various semantic IoT service are under development.

ACKNOWLEDGEMENTS

This work was supported by the project “Integrated Weather Services for Urban and Rural Area” of CATER.

REFERENCES

Semantic Web Technologies for
User Generated Content Copyright Management

Roberto García¹, Nick Sincaglia²

¹Universitat de Lleida
Jaume I 69, Lleida, 25001, Spain
rgarcia@diei.udl.cat

²NueMeta LLC
20 North Upper Wacker Drive, Chicago, USA
nsincaglia@nuemeta.com

Abstract. The growing complexity and scalability requirements of Web-wide copyright management require new approaches beyond text boxes with copyright statement, database-level flags or XML rights expressions. Semantic technologies have been successfully applied in the context of the MediaMixer European project and in collaboration with NueMeta LLC to real scenarios involving fine-grained copyright management and User Generated Content.

Keywords. Copyright, Ontology, Semantic Web, Reasoning, DAM, DRM, UGC.

1 Introduction

The European project MediaMixer promotes the adoption of semantic technologies to support organisations managing their media assets, from media fragmentation and their semantic annotation to copyright management. As part of the project demoing activities and in collaboration with NueMeta LLC, the pilot described in this paper was developed for Sony DADC, Sony’s division for digital music management and distribution.

The pilot focuses on the use of semantic technologies for copyright management of User Generated Content (UGC). UGC is content created by users and shared through platforms like YouTube. Usually, it is the mix of user content, like a wedding video, combined with content owned by others, like the couple’s favourite song by Adele.

To compensate rights holders for their content used without permission, UGC platforms provide identification services so owners can register it and be warned when it is reused. And though the first reaction might be to block content reusing copyrighted media without authorisation, UGC platforms have created a new revenue stream by sharing part of the advertisement revenue generated.

This is becoming an important income source for owners of hits in UGC. However, it requires a big mind change in media management. To foster media remixing and viral reuse of content, content owners should move away from content protection measures like DRM that might prevent their content from being remixed. They should focus on technological measures that facilitate reuse, while tracking and managing copyright, not just at the end-user level but through the whole value chain of mixes and remixes.

The solution proposed to Sony DADC is based on semantic technologies and rooted on copyright law. Thus, it provides the modelling tools to capture copy-
right statements from sources ranging from digital operations to talent contracts [1]. Moreover, thanks to the reasoning features these technologies provide, semantic rights expressions can be used to support intelligent decision support at the scale of a media repository, as detailed next.

2 Semantic Copyright Management

The proposed solution uses semantic technologies to help media owners determine, when content owned by them is detected in UGC, if they can claim control and thus block or monetize it. This is currently a complex and mainly manual task, which should take into account rights details captured by digital operations systems but also existing contracts and policies that might apply just to particular territories or include special provisions for particular artists.

For instance, the aim might be to support deciding what to do if a song by Green Day, for which some rights are hold, is detected as part of UGC. Should it be monetized and streamed advertisements supported or blocked? And, do we actually hold the required rights to claim that action in that particular situation?

With semantic technologies it is possible to go beyond just choosing to monetize based on the limited information available at digital operations. The objective is to avoid the legal troubles that might arise from ignoring, in this particular case, that Green Day’s talent contract states “we do not want our creations mixed with war images”.

The proposed solution provides a scalable decision support system capable of integrating digital rights languages, like DDEX [2] or ODRL [3], together with contracts or policies, like talent contracts or business policies. Semantic technologies provide a common and expressive framework where all these copyright information sources can be put together, the Copyright Ontology.

The Copyright Ontology goes beyond access control languages and models the core concepts in the copyright domain, starting from the different rights that compose Copyright, from Economic Rights like Reproduction Right to related rights like Performers Rights and even considering Moral Rights.

However, this is not enough to provide computers and understanding of copyright beyond being a hierarchy of rights. The aim is to model, for instance, what does it imply to hold the Fixation Right. The Copyright Ontology includes an action model that represents the different “states” a creation visits during its lifecycle and the actions that copyright value chain participants do to move creations along, as shown in Fig. 1.

A value chain is then modelled based on a subset of these actions [4]. Each of these actions is connected with corresponding right, for instance the fix action is governed by the Fixation Rights. Therefore, to hold the Fixation Right on a creation like a performance or broadcast means that it is possible to perform the fix action on it to produce a fixation. Something that then can be copied if the Reproduction Rights is also hold to produce physical copies like DVDs…
Actions are the main building block but, like sentence verbs, they need to be associated with the other entities participating in a copyright action through different facets, as shown in Table 1.

The Copyright Ontology provides a common understanding for copyright terms, but it also provides a pivoting point around which other terms can be organised and integrated. For instance, the terms used in a digital rights language like DDEX or schema.org Actions [5].

Once all these semantic building blocks are placed together, they become a really powerful and versatile way of modelling digital rights and with enough detail to make them easy to integrate and machine actionable. Using the Web Ontology Language and capable reasoners, they can be fed with semantic expressions that define patterns of actions that are allowed or prohibited. For instance, Table 2 shows a disagreement that point to an OWL class (inner box) that defines the set of ad-supported streaming actions to be prohibited, thus involving Green Day’s creations being part of content about war.

---

Table 1. Different facets to view and model entities participating in a copyright action

<table>
<thead>
<tr>
<th>Facet</th>
<th>Main role</th>
<th>Other roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Who?</td>
<td>agent</td>
<td>participant (indirect co-agent), recipient</td>
</tr>
<tr>
<td>When?</td>
<td>pointInTime</td>
<td>start, completion, duration</td>
</tr>
<tr>
<td>Where?</td>
<td>location</td>
<td>origin, destination, path</td>
</tr>
<tr>
<td>What?</td>
<td>object</td>
<td>patient (changed), theme (unchanged), result (new)</td>
</tr>
<tr>
<td>With?</td>
<td>instrument</td>
<td>medium</td>
</tr>
<tr>
<td>Why?</td>
<td>aim</td>
<td>reason</td>
</tr>
<tr>
<td>How?</td>
<td>manner</td>
<td></td>
</tr>
<tr>
<td>If?</td>
<td>condition</td>
<td></td>
</tr>
<tr>
<td>Then?</td>
<td>consequence</td>
<td></td>
</tr>
</tbody>
</table>
OWL classes definitions like the previous one are then used by reasoners to check if a particular action that some agent is trying to perform, like trying to monetize an asset by streaming it on YouTube, is allowed or not. They simply check it against all semantic models created for digital rights expressions, policies or contract about that asset.

As shown in Fig. 2, reasoners classify intended actions in class patterns and then check if they correspond to agreements and there is no disagreement prohibiting them. Therefore, they do the hard work of checking all possibilities, minimizing implementation costs while maintaining the flexibility and scalability of the proposed solution.

Fig. 2. Reasoner classification of two intended actions (black dots) against an agreement from a DDEX Deal and a disagreement clause in Green Day’s contract. Though the former might authorize both, the latter blocks YouTube streaming within UGC from happening.

References

2. Digital Data Exchange (DDEX), http://www.ddex.net
3. Open Digital Rights Language (ODRL), http://www.w3.org/community/odrl
RDF Implementation of Clinical Trial Data Standards

Frederik Malfait¹ and Josephine Gough²

¹IMOS Consulting, Switzerland
²F. Hoffmann-La Roche AG, Switzerland

frederik.malfait@imos-consulting.com, josephine_anne.gough@roche.com

Abstract. Clinical trials pose increasing challenges to sponsors and regulators in terms of execution complexity, timing constraints, risk management, and quality of scientific output. We show how the systematic use of clinical trial data standards, combined with the application of model driven semantic technology within the context of a metadata registry can provide a more effective way to start addressing these challenges. This approach has been implemented and deployed as a validated production system within a large pharmaceutical company.

Keywords: clinical trial data standards, metadata registry, model driven semantic technology

1 Business Context

The planning, execution, and analysis of clinical trials has many touch points throughout any large pharmaceutical company. The reference document of a clinical trial is the clinical study protocol, but being (usually) a paper document, there are many hand-offs and labor intensive tasks in downstream processes to properly define, collect, tabulate, analyze, and submit clinical trial data to regulatory authorities. This touches on virtually every aspect of the clinical trial data life cycle and incurs significant cost.

Progress has been achieved by increased adoption of clinical trial data standards, most notably the leading industry standards that are developed and published by the Clinical Data Interchange Standards Consortium (CDISC). Many pharmaceutical companies implement at least some of these standards. There are also an increasing number of data standard groups to support standards implementation and governance within a pharmaceutical company.

By themselves data standards are useful, but still lack key features to achieve their full business value. Clinical trial data standards often lack a consistent background model, are published in PDF or other office document formats, are inconsistent across different areas, have insufficient versioning capabilities, and are difficult to reference, maintain, extend, and integrate with other information resources.

The issue of a consistent background model for data standards can be addressed by adopting the ISO 11179 Metadata Registry (MDR) standard. This standard introduces common terminology and a basic meta-model to define, register, and version data elements and related items in a metadata registry (MDR). W3C semantic technology standards such as RDF, RDFS, OWL, and SKOS close the gap by introducing a standard language that can express an ISO 11179 type meta-model and layer data standard model instances on top of that.

At Hoffmann-La Roche, an ISO 11179 metadata registry has been designed, implemented, and deployed using semantic technology. The system has gone through several releases, making increased use of semantic capabilities based on a model driven approach. Each release is fully validated and complies with the US Food and Drug Administration (FDA) 21 CFR Part 11 that regulates computer system validation. The system is currently used to manage and serve access to the Roche clinical trial data standards. Future releases are planned to implement standards driven workflow
automation to optimize current processes for collecting, tabulating, analyzing, packaging, and submitting clinical trial data and results to regulatory authorities.

2 Information Model

The first release in 2011 implemented a fully standards based information model for the definition, management and dissemination of clinical trial data standards throughout its life cycle as shown in Figure 1.

- Sponsor Extensions
- CDISC Foundational Standards
- ISO 11179 model for Metadata Registries (MDR)
- W3C Semantic Standards RDF - RDFS - OWL - SKOS

Figure 1 RDF Implementation of a Clinical Trial Data Standards Registry

The information model exploits standards at all levels. RDF is used as a standard language to define and link all meta-models and models needed to express the clinical trial data standards. The core of the ISO 11179 standard provides a concise vocabulary to express metadata models and defines a unified background model to express clinical trial data standards. CDISC is the leading body that defines industry wide clinical trial data standards, ranging from data collection (CDASH) and data tabulation (SDTM) to data analysis (ADaM). Finally, as science and standards evolve, there will always be a need to also introduce sponsor specific standards.

All this information is ultimately expressed in RDF, which acts as a universal language to structure and layer the different levels of the information model. Using linked data principles, it also provides an excellent capability to interconnect the different models and enable traceability of the clinical trial data standards from protocol to regulatory submission.

3 Model Driven Architecture

Information is disseminated through a REST based API that serves data to a browser application, a search facility, and other standards driven applications. Increased use of the data standards repository led to further demand for deploying additional schemas, new data sources, and extensions of the existing REST interfaces. Being a validated application, this turned out to be a very costly process. The fourth release of the system deployed in 2013 addressed this by introducing the capability for model driven services that are themselves expressed in RDF.
The implementation of model driven services is based on facet descriptions of RDF resources. A REST request for a resource invokes an engine to process a facet that exposes a configurable bounded description of an RDF resource. Facets themselves are written in RDF using a small footprint ontology that describes facets in terms of facet elements. Each facet element specifies a SPARQL property path starting from the requested resource together with instructions on how to return the data to the response. The property path of a facet element can point to a literal value to be passed to the response or to a set of target resources that may have their own facet descriptors. This mechanism allows recursive facet composition to describe resources in terms of their own properties and any dependencies they may have on other resources. It is possible to define multiple facets for the same resource and thus expose the same resource using different representations depending on application needs. Integrated with an XSLT engine, the facet mechanism supports many different output formats such as XML, JSON, PDF, Word, and Excel.

Currently validated and deployed are services that include exposure of resources for GET requests, specification of searchable facets, data validation rules expressed as SPARQL ASK queries, and the option to create model driven user interfaces. Configurable services enable the publication of new or updated RDF schemas and data sources, bundled together with RDF configurations that define how to access, search, view, and consume these data sources.

4 Future Work

Currently being developed are services for model driven PUT/POST/DELETE requests and a model driven security framework. Deployment of these additional model driven services is planned for the end of 2014.

A model driven architecture as described above enables a more flexible approach to develop applications on top of the REST service layer. Use cases include a computable protocol representation, component based authoring, definition of the study schedule (activities and visits matrix), and downstream systems automation for setting up clinical trial databases and generating clinical trial submission data sets. The requirements phase for these applications has been concluded and implementation is envisaged for 2015.

5 Industry Initiatives and Collaborations

Just like with the web and RDF itself, the value of clinical trial data standards increases significantly if the benefits of the network effect can be realized. Standards become more valuable with increased adoption. To this end, Roche has shared many of the RDF models as a starting point to represent most the CDISC foundational standards in RDF in several industry collaborations. This work is pursued mainly by the Semantic Technology (ST) working group of the PhUSE Computational Science Symposium (CSS), in close collaboration with CDISC subject matter experts.

In 2013, work was completed for the RDF representation of a series of CDISC foundational standards, including CDASH, SDTM, SEND, and ADaM, which are planned to become subject for CDISC public review in 2014. Currently, PhUSE CSS teams in the Semantic Technology group are working on a protocol representation model in RDF, analysis metadata representation in RDF (possibly using the RDF Data Cube Vocabulary), linking Electronic Health Records (EHR) with CDASH data collection standards (KeyCRF project), and representing regulatory guidelines in RDF.

References

1. CDISC. Analysis Data Model (ADaM). http://www.cdisc.org/adam.
Fast In-Memory Reasoner for Oracle NoSQL Database EE: Uncover Hidden Relationships that Exist in Your Enterprise Data

Zhe Wu\textsuperscript{1}, Gabriela Montiel\textsuperscript{2}, Yuan Ren\textsuperscript{3}, and Jeff Z. Pan\textsuperscript{3}

\textsuperscript{1}Oracle, US
\textsuperscript{2}Oracle, Mexico
\textsuperscript{3}Department of Computing Science, University of Aberdeen, UK

Graph databases and NoSQL databases, two very important topics in Big Data, have gained popularity in recent years due to their unique characteristics in their horizontally scale-out capability and flexible schema or schema-free design. The recent release of OWL-DBC\textsuperscript{1}, an adaptor between Oracle Spatial and Graph\textsuperscript{2} and the TrOWL reasoner [2, 1], has built a tight integration between one of the leading industrial graph databases and the cutting edge, in-memory, semantic reasoner to achieve high quality and efficient semantic reasoning on large scale enterprise data.

In this session we present OWL-NOSQL, which enhances the Oracle NoSQL Database EE\textsuperscript{3} with efficient in-memory reasoning capability from TrOWL. With OWL-NOSQL, users are able to manage their enterprise data in the form of RDF Graph stored in Oracle NoSQL Database EE and gain insight into their data through powerful semantic reasoning.

Oracle NoSQL Database EE is a horizontally scaled, key-value database for Web services and cloud. This system uses a simplistic key-value pair data model to achieve efficiency and high scalability. Despite of its simplicity, such a data model can be engineered to represent rather complex knowledge and structures in data, including RDF graphs and OWL ontologies. In fact, key-value pair databases have emerged as one of the promising solutions for semantic exploitation in recent years. Such flexibility enables Oracle NoSQL Database EE to expose its data to external semantic applications, including semantic reasoners, to uncover hidden relationships in the stored data, especially those represent semantic annotations. More concretely, such a semantic extension of Oracle NoSQL Database EE is performed as follows:

1. Exporting RDF data stored in Oracle NoSQL Database EE into an ontology.
2. Performing reasoning using the semantic reasoner TrOWL to uncover hidden relationships in the data.
3. Importing reasoning results into Oracle NoSQL Database EE to persistent the uncovered relationships.

\textsuperscript{1}http://trowl.eu/owl-dbc/
\textsuperscript{2}http://www.oracle.com/technetwork/database/options/spatialandgraph/overview/index.html
\textsuperscript{3}http://www.oracle.com/us/products/database/nosql/overview/index.html
According to our previous experience with OWL-DBC, the most significant performance challenge of such a solution rises from the data transferring between database and reasoner. To address such an issues and offer faster data exploitation, the following means have been taken:

1. Performing reasoning and data export/import in memory. This minimises the need to perform storage I/O.
2. Enable parallel processing of data. Such parallelisation can be realised on two levels:
   (a) Export, reasoning and import can be performed in parallel to each other. Particularly, once a reasoning result is being computed, it can be directly imported into Oracle NoSQL Database EE, without having to wait for the other reasoning results. Such parallelisation exploits the parallel mechanism between storage, memory and CPU cores. It is even possible for TrOWL to start reasoning without waiting for all data being exported from Oracle NoSQL Database EE. Nevertheless, it is worth noting that such further parallelisation may have an impact on the completeness of results when inference requires global condition checking. Hence, it should be applied with carefully design reasoning procedure and modularisable dataset.
   (b) Export, reasoning and import individually can be performed in parallel. Particularly, Oracle NoSQL Database EE is capable of exporting and importing data in parallel using multiple threads. TrOWL, on the other hand, is capable of executing reasoning on several mutually independent partitions. On a computer (cluster) with multiple storage I/O bandwidth and multiple CPU cores, such parallelisation can make the best use of all available hardware resources.

With the above solutions, we are able to improve the efficiency of data transferring and reasoning. Together, OWL-NOSQL enhances Oracle NoSQL Database EE with semantic reasoning, offering more flexibility to our clients in terms of data storage, management and exploitation options.

We are optimistic about OWL-NOSQL because many industries have already embraced Semantic Web and NoSQL technologies. In the past decade, we have observed more and more enterprise applications built on top of RDF and OWL standards; and NoSQL technologies at the same time play an ever-increasingly critical role for the management and analysis of Big Data. There clearly is a natural synergy between these two sets of technologies.

References

Semantic Technology for Oil & Gas Businesses

Nico Lavarini
Expert System SpA, Rovereto – Trento, Italy
nlavarini@expertsystem.com

Abstract. Global events can trigger new challenges and changes in strategy in the blink of an eye. Knowledge intensive industries require early and accurate notification of global market changes as well as solid information collection strategies which are vital to good decision making. For these purposes, the adoption of semantic technology will make companies aware of changes (ongoing and prospective) ahead of time: a key factor in staying competitive and up to date. The Cogito Semantic Intelligence Platform uses semantic technology to reach unprecedented levels of depth and quality in intelligence analysis. The software drastically reduces the amount of manual analyst labor while providing immediate and complete information to aid multi-national Oil & Gas Companies in the decision-making process.

Keywords: Semantics, semantic technology, Oil & Gas, search, discovery, extraction, classification, analytics, unstructured information, knowledge management, competitive intelligence, decision making process

1 The scenario and its challenges

International Oil & Gas companies are committed to finding, producing, transporting, transforming and marketing oil and gas.

Within these companies, there are corporate functions devoted to Integrated Initiatives Promotion, where “integration” implies the achievement of core business results across several stages of the hydrocarbon value chain (upstream and downstream).

Creating value in today’s globalized business encompasses a tight and continuous effort in understanding and anticipating what’s to come environmentally, economically, politically or otherwise. As such, technological innovation is crucial to stay afloat in the energy and fuel market.

The risk for any international player is having an intelligence process that is slower than the speed of destructive marketplace events or having an “after the fact” approach in the process of tackling risks and opportunities in business and technology environment, with little or no room for maneuvering.
2 The solution

The adoption of semantic technology has enabled knowledge workers to select and connect pieces of knowledge critical to maintaining stability in a tumultuous industry. Field experience shows that knowledge is more about the connections among concepts within a text rather than just a mere collection of keywords or documents. A semantic engine is the only tool that can improve productivity and effectiveness in making such connections.

Cogito™ semantic software is that engine. Oil & Gas knowledge workers can employ such a semantic search engine to

- Capture the weak signals and nuanced technological advancements buried in textual documents;
- Distill key information, for example using SAO (Subject, Action, Object) structure identification, to boost technical problem solving so informed decisions can be made;
- Select, store and track the signals of knowledge from the huge overflow of current information until they become large enough to be classified as an opportunity or threat.

2.1 Key benefits

The benefits of adopting advanced semantic search tools range from the effectiveness of searching – providing targeted responses and extracting “meaning” from text documents – to the more complete organization of the company’s information assets. It also ensures security and compliance, fosters knowledge sharing and cooperation among all users, and gives tangible support for business processes that are developed in the pursuit of the strategic mission.

- Reduced time and costs for information management using data mining, semantic search, content navigation and automatic categorization
- Monitoring of markets and competitors for competitive advantage
- Advanced collaboration platform for knowledge workers and analysts to share intellectual capital

The Cogito engine allows operators to manage information by capturing the weak signals in content, storing and tracking knowledge in an R&D environment, and distilling key analytical data.

3 Case studies

Scientific breakthrough: A government spokesman announces in the early morning that a scientist from Addis Ababa, Ethiopia, was able to produce hydrogen using bacteria. With the markets in London set to open in three hours, African press agencies start to push out news of the discovery.
Meanwhile, a knowledge worker at the headquarters of the Oil & Gas Group is just turning on his PC and opening the Cogito application from Expert System. On the map, the knowledge worker notices a red light indicating a breakthrough in the African region. The worker begins analyzing information related to the discovery and is quickly able to notify his superiors of the implications for the Company — which are monumental — since they are looking for new and innovative ways to produce hydrogen.

**Political and geographical awareness:** Regional Authorities in the Gansu region of China are discussing the possibility of a new National Park area which would cover a significant part of the region. This information is still largely unknown to main press agencies, as it is still a just a simple proposal.

A Company has plans for an important gas plant in the area, and thus, is seeking political collaboration from local authorities. The information mentioned above could drive the Company strategy away from the area, or at least change the mid-term strategy. Knowing this type of information ahead of time could give the Company a lead on its competitors in the race for worldwide business.

**Knowledge on territory:** A Group spans almost a hundred countries, with tens of thousands employees. The amount of available documentation is huge, and sorting millions of documents for relevant information is a complex matter.

The Cogito platform performs semantic categorization and content identification, so that large unstructured document bases can be managed in a structured way, and thus extracts relevant documents in just seconds. For example, an operator using Cogito can effectively distill the large document base and extract only the documents ‘relevant to the Gansu region which discuss Hydraulic Fracturing strategies which include the collaboration of sub-contractor EOG Resources’, without even knowing the exact keywords and terms used in the various sources contained in the document base.

**Corporate Knowledge Management:** A Group manages several million technical documents containing deep technical content related to the Oil & Gas domain. The geographical aspects of documents (locations of plants, basins, oil fields, all proprietary and generic locations) are of extreme importance. The entire database of documents is easily processed by the Cogito platform, which contains a rich customized semantic network of millions of geographical entities and their locations, including their relationships to other locations and resources.

The processed documents are then enhanced by Cogito with a geographic bounding box (a coordinate rectangle containing the relevant locations and resources in the document). These bounding boxes can be used to instantly identify all the documents relevant to any area worldwide, and selection can be further filtered by topic, category, contained entities, etc. In this way, the Group can immediately have all and only those documents strictly related to any geographical location in the world, ranked by distance, relevance to topic, etc.
Social and political awareness: Websites providing breaking news announce uprisings in the Gombe State of Nigeria. The Company has a major oil plant in this region so Analysts are keeping this area (and others) monitored with the Cogito platform.

News of the uprisings is “pushed” to the operators without user action, thus notifying the Company as soon as news is available. This gives the Company the opportunity to immediately deploy evacuation or other security-related procedures for the staff and families working in the plant, before the staff itself is even aware of the uprising.

3.1 The facts

While some particulars of these stories are fictional, the overall narratives are not. Global Oil & Gas Companies face situations similar to these on a daily basis. These companies are present in many countries, and world events can trigger new expenditures and changes in strategic direction. To address these unique challenges and benefit from early and accurate notification, the solution is an investment in an innovative business intelligence platform.

Companies have turned to Expert System’s Cogito platform for an especially precise tool to help them improve the information flow from external sources, enhance data management and make the best use of the intellectual capital within the company.

4 The future

Semantic technology has played, and will continue to play, a critical role in supporting strategic decision processes, handling data in internal knowledge generation and management, and monitoring and detecting early signals of change in the energy markets.

For the future, Expert System customers can expect that this collaborative investment in semantic intelligence will contribute to the enhancement of the knowledge management process in other parts of the company worldwide and solve the problems of

- Wasting efforts on poor content, through an early selection of high quality information without “fully reading” the text.
- Not finding/receiving information that could be potentially detrimental to the business.
- The unnecessary re-creation of content, because managing knowledge with semantics overcomes the “If we only knew what we know” syndrome.
keyCRF: Using Semantic Metadata Registries to Populate an eCRF with EHR Data

Gokce B. Laleci Erturkmen1, Landen Bain2, Anil Sinaci1, Frederik Malfait3, Geoff Low4

1 SRDC Ltd., Ankara, Turkey
2 CDISC, USA
3 ROCHE, Switzerland
4 Medidata Solutions, London, UK

{gokce, anil}@srdc.com.tr, lbain@cdisc.org, frederik.malfait@roche.com, glow@mdsol.com

Abstract. The goal of the keyCRF project is the creation of a semantically annotated electronic Case Report Form (eCRF) that can enable the pre-population of the eCRF from data elements in an EHR summary document through the use of semantically linked Common Data Element definitions across care and research domains.

Keywords: semantic metadata registry, re-use of EHR data, eCRF, Common Data Elements

1 Introduction

A major barrier to repurposing clinical data of electronic health records (EHRs) for clinical research studies (clinical trial design, execution and observational studies) is that information systems in both domains – patient care and clinical research – use different data models and terminology systems. Different data representation standards, and Common Data Element (CDE) models are being used to facilitate seamless data exchange between disparate systems [1].

The Clinical Data Interchange Standards Consortium (CDISC) provides common dataset definitions in (a) Study Data Tabulation Model (SDTM) [2] for enabling the submission of the result data sets of regulated clinical research studies to the FDA and in (b) Clinical Data Acquisition Standards Harmonization (CDASH) [3] for integrating SDTM data requirements into the Case Report Forms. On the care site previously Health Information Technology Standards Panel (HITSP) has previously defined the C154: Data Dictionary Component [4] as a library of data elements. HITSP C32 [5] which describes the HL7/ASTM Continuity of Care Document (CCD) content for the purpose of health information exchange, marks the elements in CCD document with the corresponding HITSP C154 data elements to establish common understanding of the meaning of the CCD elements. Later as a part of Meaningful Use Stage 2, Consolidated CDA (C-CDA) templates are provided for enabling exchange of patient clinical data[6]. The Transitions of Care Initiative (ToC) maintains the S&I Clinical Element Data Dictionary (CEDD) [7] as a repository of data elements in support of meaningful use and improvement in the quality of care. As the data exchange formats and common data elements provided by these two domains are different, it is not automatically possible to pre-fill an electronic case report form annotated with CDISC SDTM and CDASH variables by re-using the medical history of a patient available in a C-CDA document.

The keyCRF project [8] initiated as a part of PhUSE Computational Science Symposium Semantic Technology workgroup, aims to facilitate this, through a metadata registry that maintains the semantic links between the common data elements used in research and care domains and through the use of IHE Data Element Exchange (DEX) [9], and IHE Retrieve Form for Data Capture (RFD)[10] profiles collaboratively.
2 Methods and Expected Results

One of the core activities of keyCRF project is to identify a sample set of CDEs in research sites that are often used to annotate eCRF forms, and the corresponding CDEs at clinical care site, and semantically linking them through a semantically enabled metadata registry implemented in conformance to ISO/IEC 11179 standard. We will be using the semantic MDR implementation provided by SALUS project [1] that enables mapping of CDEs managed by different domains through skos terms such as skos:exactMatch and skos:closeMatch. The activities being carried out can be summarized as:

- Examine the sample eCRF form provided by CDISC and identify the CDEs at research sites from the common CDASH and SDTM annotations of eCRF forms, such as “DM.SEX” to indicate gender code in demographics domain.
- Examine CEDD repository, to find out the corresponding CDEs to the selected research CDEs, for example “PatientInformation.PatientAdministrativeGender.CE” to “DM.SEX”.
- Represent all these CDEs in a 11179 supporting MDR, by also semantically linking them with skos terms.
- Define extraction specifications of the selected CEDD CDEs from C-CDA as XPATHS.

By making use of these definitions available from a semantic MDR that also supports IHE DEX as a standard means to retrieve metadata of CDEs, we demonstrate that it is possible to pre-fill an electronic case report form by reusing the medical history available as follows: A research forms designer becomes able to build a case report form for a particular research study by referring to an on-line metadata registry of research data elements, and selects the desired data elements from a set of research friendly elements such as CDASH. He then retrieves the metadata defined by the metadata registry into an annotated case report form through the use of IHE DEX profile. The metadata includes the exact specification, using XPath, to find the corresponding data element in the C-CDA. The semantic MDR creates the metadata by checking the semantic links of CDASH data elements to CEDD data elements, which already have mappings to C-CDA documents. Using the XPath statements, the research system creates an extraction specification for all elements to be extracted from the C-CDA. The demonstration will employ the well-known mechanism of IHE RFD to define the necessary transactions between the EHR and the research system. The extraction specification could then be used with IHE RFD to pre-populate the case report form.

This prototype demonstration will show industry the value of the semantic approach to address the challenge of secondary use of EHR for research purposes.

References

2. CDISC. Study Data Tabulation Model (SDTM), http://www.cdisc.org/sdtm
3. CDISC. Clinical Data Acquisition Standards Harmonization (CDASH),http://www.cdisc.org/cdash
Building and Exploring Marine Oriented Knowledge Graph for ZhouShan Library

Tong Ruan¹,², Haofen Wang¹,², Fanghuai Hu², Jun Ding², and Kai Lu³

¹ East China University of Science & Technology, Shanghai, 200237, China
{ruantong,whcarter}@ecust.edu.cn
² Shanghai Hi-knowledge Information Technology Corporation
{hufh,dingjun}@hiekn.com
³ ZhouShan Library, Zhe Jiang Province, China
{zslukai}@126.com

Abstract. Thematic repositories targeting to local economy or regional culture are becoming landmarks of regional digital libraries in China. Customers from these libraries are searching for advanced technologies to improve their repositories from all aspects. On the other hand, search engine companies like Google and Baidu are building their own knowledge graphs (KG) to empower the next generation of Web search. Due to the success of knowledge graphs in search, regional libraries are eager to embrace KG-related technologies. In this paper, we give a introduction on how to build a marine-oriented repository for ZhouShan library with KG technologies, and related business and technique problems encountered are discussed as well. We also present an integrated tool VKGBuilder to help users to manage the life cycle of marine-oriented knowledge graph. At last, we give a plan on a big connected KG operated by different libraries in the near future.

1 Paradigm Shift for Library Industry in China

As more and more readers are in favour of accessing digital resources online, most libraries in China are in their way to build or strengthen their digital libraries. Nowadays, there exist several major content providers like WeiPu⁴, WanFang⁵, and ChaoXing⁶ who not only own a large number of digital contents of journals, books, and magazines, but also run their integrated platforms for search and navigation. Most libraries only act as a consumer or a distributor in the digital content supply chain, which makes them suffer from serious homogenization, lack of content control, and weak competitiveness. The above issues enforce libraries to search for new opportunities.

Figure 1 shows the current situation of library industry in China. Libraries from different regions and of different levels buy similar resource from content

⁴ http://oldweb.cqvip.com/
⁵ http://www.wanfangdata.com.cn/
⁶ http://www.chaoxing.com/
providers. For example, ZhouShan is a city of ZheJiang province, and both Zhe- 
Jiang provincial library and ZhouShan library buy contents from ChaoXing and 
WanFang, as shown in Figure 1. In most cases, content providers run their own 
web-based knowledge service systems, and library portals contain links to con-
tent provider’s portals. Consequently, content providers own resource-level user 
access logs and can provide value-added services such as document recommend-
ation. So Local readers in ZhouShan would prefer national digital library since 
these libraries have more resources, and they would prefer content provider’s in-
tegrated platforms for better services. In both cases, the municipal or provincial 
digital libraries have no competitive advantages.

On the other hand, in early 2013, China Ministry of Culture has issued 
guidelines to build various resource repositories specified for different sectors. 
It advocated different regions to develop thematic repositories according to the 
economic and cultural characteristics of the region. ZhouShan library takes this 
chance and becomes a pioneer to make the transition. ZhouShan Islands are 
listed as the first “state-level new district” around marine economy. With the 
support of local government, ZhouShan library starts a project named “Universal 
Knowledge Repository for Marine Digital Library”. The intention is to help 
habitants and travellers know ZhouShan and marine economy, and to support 
different bureaus of ZhouShan government, such as Fishery Agency or Economic 
and Information Commission to do queries and statistics about local marine 
economy. In this way, ZhouShan library is changing from a content distributor 
to a content provider of the marine domain. This change also happens to other 
regional libraries, which leads to a trend of paradigm shift in China’s library 
industry.

![Fig. 1. Current paradigm of library industry in China](image)

2 The Role of (Vertical) Knowledge Graph

**Business and Technical Requirements** Regarding the project of ZhouShan 
library, customers themselves have neither enough marine materials with copy-
rights nor human resources to construct a marine repository. They have to rely 
on an open platform which could utilize data from all possible data sources and 
could leverage enthusiasts to extend data. They also expect distinguished end 
user experiences that differ their repositories from traditional ones. Technically, 
the platform should have the following capabilities: a) *Integrate data from multiple and diverse data sources.* A marine repository should include fishes, fishing 
grounds, fish processing methods, related researchers and local enterprises. No
single source can cover all aspects of data in the repository. It is also impossible for users to manually integrate knowledge from various sources. In some cases, concepts or facts need to be extracted from semi-structured data (e.g., lists or tables from Web pages) and unstructured data (e.g., documents). In other cases, data from internal databases or from LOD (Linked Open Data) are to be extracted, transformed, and loaded to the repository in a unified representation.

b) Incremental data update. Research institutes, government bureaus, and marine enthusiasts alliances are allowed to continuously add new knowledge to make the repository up-to-date.

**Reasons to Choose KG Technology** To fulfill the above requirements of repository construction, ZhouShan library embraces semantic technologies to build a vertical marine oriented knowledge graph (see Figure 2). Knowledge Graph (KG) was first introduced by Google to empower its search. The big success of knowledge graph attracts many attentions from other internet companies as well as traditional industries. The main advantages of semantic technologies include: a) Incremental schema design and enrichment. It is difficult to know all concepts during the initial design of KG. Its dynamic extensibility and “schema-less” characteristic enable to add new schemata or revise existing ones later without rebuilding the whole KG from scratch. b) Easy data integration. The semantic interoperability of ontologies and the “linked data” principle make it

---

7 http://linkeddata.org
more efficient to integrate digital contents from different content providers. c) *Existing standards support.* The library can urge the content providers to obey the existing standards like URI, RDF(S), and SPARQL. d) *Expressive semantic search.* Users can ask for entities satisfying semantic constraints when searching the KG, which is more precise than keyword-based retrieval.

3 Deployment of KG in ZhouShan Library

ZhouShan Marine Digital Library based on marine-oriented KG is online this September. End users can search the marine repository and related books with natural language queries. They can also browse marine-related entities with 3D animation. Marine experts can edit the repository via a collaborative editing interface. The newly built library also provide restful APIs for bureaus of local government and enterprise to retrieve marine data for decision making. The construction process of the repository only requires limited human interventions because we build an integrated tool VKGBuilder to help KG construction.

VKGBuilder has three key components namely *knowledge integration* module, *knowledge store* module, and *knowledge access* module. As for knowledge integration, *knowledge integration* module includes functions such as converting relational data from internal databases to RDF triples, extracting facts from user generated contents like Wikipedia, importing marine related ontologies from the Web, and fusing data at both schema level and data level. Moreover, there are build-in rules to detect schema inconsistencies and data conflicts, and collaborative editing tools for users to extend and validate KG. For the design of *knowledge store,* it is a virtual graph database which combines RDBs, in-memory stores and inverted indexes to support fast access of KG in different scenarios. Regarding *knowledge access* module, there are different ways including card view, wheel view, and detail view to navigate and browse KGs. Besides, semantic search which converts natural language queries to SPARQL is supported as well. Finally, *knowledge access* module also contains a list of restful APIs for developers to interact with the underlying KG.

4 Federated KG in the Future

Other regional libraries such as ZheJiang provincial library and Shanghai library also want to build their own thematic repositories. The resulting KGs from different libraries may be complementary. We envision a federated KG with new nodes added dynamically, as shown in Figure 3. The federated KG would show more benefits of semantic technology such as KG or Linked Data, as these technologies are born with "Link" in mind. There remain problems about how to link similar concepts and entities in different KGs, or how to provide a global query interface. We will try to tackle these problems in our future work.

**Acknowledgements** This work is funded by the National Key Technology R&D Program through project No. 2013BAH11F03.

* You can access the production web site via [http://kd.zsodl.cn](http://kd.zsodl.cn).
Traffic Management using RTEC in OWL 2 RL

Bernard Gorman, Jakub Marecek, and Jia Yuan Yu

IBM Research – Ireland
B3 Damastown Technology Campus, Dublin 15, Ireland
{berngorm,jakub.marecek,jiayuanyu}@ie.ibm.com

Introduction

In a number of domains, including traffic management, event processing and situational reporting are particularly demanding. This is due to the volumes and reliability of streamed spatio-temporal data involved, ranging from sensor readings, news-wire reports, police reports, to social media, as well as the complexity of the reasoning required. Human, rather than artificial, intelligence is hence still used to an overwhelming extent.

A number of specialised event-processing languages and reasoners have been proposed, extending RDF and SPARQL. These include SPARQL-ST [13], Temporal RDF [18] and T-SPARQL [8], Spatio-temporal RDF and stSPARQL [10]. For even more elaborate extensions, see e.g. [14, 2, 11]. Often, these extensions rely on custom parsers for the languages and on custom Prolog-based implementations of reasoners. Yet, none of these extensions has gained a wide adoption.

We argue that such specific languages and reasoners go against the principle of a general-purpose description logics and general-purpose reasoners [3]. We propose a rewriting of RTEC, the event processing calculus [2], from Prolog to OWL 2 RL [9], which is the only profile of the Web Ontology Language, for which there exist very efficient reasoners [12, 16, 15].

RTEC

Artikis et al. [2] proposed Event Calculus for Run-Time reasoning (RTEC) as a calculus for event processing. Prolog-based implementations, where event processing is triggered asynchronously and the derived events are produced in a streaming fashion, are readily available [1]. In order to make this paper self-contained, we summarise its principles beyond the very basics [7].

Time is assumed to be discretised and space is represented by GPS coordinates. All predicates in RTEC are defined by Horn clauses [7], which are the implications of a head from a body, \( h_1, \ldots, h_n \leftarrow b_1, \ldots, b_m \), where \( 0 \leq n \leq 1 \) and \( m > 0 \). All facts are predicates with \( m = 0 \) and \( n = 1 \), such as \( \text{move}(B_1, L_1, O_7, 400) \), which means that a particular bus \( B_1 \) is running on a particular line \( L_1 \) with a delay of 400 seconds, as operated by operator \( O_7 \). Similarly, \( \text{gps}(B_1, 53.31, -6.23, 0, 1) \) means that the bus \( B_1 \) is at the given, its direction is forwards \((0)\) and there is congestion \((1)\). Based on such facts, one formulates rules, i.e. Horn clauses with \( m > 0 \) and \( n = 1 \), for the processing of instantaneous events or non-instantaneous fluents. The occurrence of an event \( E \), which is an inferred Horn clause with \( m > 0 \) and \( n = 1 \), at a fixed time \( T \), is given by rules using \( \text{happensAt}(E, T) \). The occurrence of a fluent \( F \) is at a finite list \( I \)
Table 1. Main predicates of RTEC. Cited loosely from [1].

<table>
<thead>
<tr>
<th>Predicate</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>happensAt(E, T)</td>
<td>Event E occurs at time T</td>
</tr>
<tr>
<td>holdsAt(F=V, T)</td>
<td>The value of fluent F is V at time T</td>
</tr>
<tr>
<td>holdsFor(F=V, I)</td>
<td>List I of intervals for which F = V holds</td>
</tr>
<tr>
<td>initiatedAt(F=V, T)</td>
<td>Fluent F = V is initiated at T</td>
</tr>
<tr>
<td>terminatedAt(F=V, T)</td>
<td>Fluent F = V is terminated at T</td>
</tr>
<tr>
<td>relative_complement_all (I0, L, I)</td>
<td>List I of intervals is obtained by complementing ( i \in I_0 ) within ground set ( L )</td>
</tr>
<tr>
<td>union_all(L, I)</td>
<td>List I of intervals is the union of those in ( L )</td>
</tr>
<tr>
<td>intersect_all(L, I)</td>
<td>List I of ints. is the intersection of those in ( L )</td>
</tr>
</tbody>
</table>

of intervals, is given using \( \text{holdsFor}(F=V, I) \). Simple fluents, which hold in a single interval, are given by \( \text{initiatedAt}(E, T) \) and \( \text{terminatedAt}(E, T) \). For an overview of the predicates, please see Table 1.

Notice that Horn clauses can be used to define complex events, such as the sharp increase in the delay of a bus parametrised by thresholds \( t, d \) for time and delay:

\[
\text{happensAt} \left( \text{delayIncrease}(\text{Bus}, X, Y, \text{Lon}, \text{Lat}), T \right) :- \\
\text{happensAt}(\text{move}(\text{Bus}, _, _, \text{Delay0}), T_0), \\
\text{holdsAt}(\text{gps}(\text{Bus}, X, Y, _, _)\text{=}true, T_0), \\
\text{happensAt}(\text{move}(\text{Bus}, _, _, \text{Delay}), T), \\
\text{holdsAt}(\text{gps}(\text{Bus}, \text{Lon}, \text{Lat}, _, _)\text{=}true, T), \\
\text{Delay} - \text{Delay0} > d, \\
0 < T - T_0 < t
\]

where comma denotes conjunction, \( _\) is the anonymous variable, and \(-\) denotes implication.

The complex events can be processed in a custom Prolog-based implementation [1], or as we show later, a OWL 2 RL reasoner [20]. In the Prolog-based implementation, one rewrites the inputs as facts, and leaves the reasoning about \( \text{delayIncrease} \) up to a Prolog interpreter. The resulting interactions between the ontology tools, Prolog interpreter, and rewriting among them are frail and challenging to debug, though.

RTEC in OWL 2 RL It has long been known that Horn clauses can be rewritten into and queried in OWL 2. Recently, it has been shown [19] that Horn clauses can be rewritten in OWL 2 RL, a tractable profile of OWL. This rewriting allows for sound and complete reasoning, c.f. Theorem 1 of [20]. Moreover, the reasoning is very efficient, empirically.

The rewriting of Zhou et al. [20] proceeds via Datalog\( ^{=,\vee} \) [5] and Datalog\( ^{\leq,\vee} \) proper into OWL 2 RL. Instead of goals in Prolog, which are Horn clauses with \( m > 0 \) and \( n = 0 \), one uses conjunctive queries in OWL 2 RL. Formally, Datalog\( ^{\leq,\vee} \) has first-order sentences of the form \( \forall x \exists y \text{ s.t. } C_1 \land \cdots \land C_m \leftarrow B \),
where $B$ is an atom with variables in $x$, which is neither $\bot$ nor an inequality. Conjunctive query (CQ) with distinguished predicate $Q(y)$ is $\exists y \phi(x,y)$ and $\psi(x,y)$ a conjunction of atoms without inequalities. In the example above, the Datalog rule is:

$$
\exists T', D, D' \{ \exists a, b (\text{happensAt}(\text{move}(\text{Bus}, a, b, D'), T')) \wedge \\
\exists c, d (\text{holdsAt}(\text{gps}(\text{Bus}, X, Y, c, d)=\text{true}, T')) \wedge \\
\exists e, f (\text{happensAt}(\text{move}(\text{Bus}, e, f, D), T)) \wedge \\
\exists g, h (\text{holdsAt}(\text{gps}(\text{Bus}, \text{Lon}, \text{Lat}, g, h)=\text{true}, T)) \wedge \\
D - D' \leq d \land \\
0 < T - T' < t \}
$$

\[\leftarrow \text{happensAt}(\text{delayIncrease}(\text{Bus}, X, \text{Lon}, \text{Lat}, T), \right]

where all free variables (Bus, X, Y, Lon, Lat, T) are universally quantified.

Following this line of work [19], we rewrite RTEC into OWL 2 RL. This is the first ever translation of RTEC into OWL 2 RL, as far as we know. In a companion paper co-authored with the staff at Dublin City Council [1], we describe an extensive traffic management system, where we employ RTEC in traffic management.

**Conclusions** The value and scalability of spatio-temporal event processing over streaming data has been demonstrated a number of times [17, 6, 1]. Notice, however, that there remains a considerable gap between first prototypes specific to a particular city and a general-purpose methodology or tools. General-purpose reasoners using RTEC in OWL 2 RL may lack the performance of custom-tailored reasoners [4], capable of dealing with gigabytes of data at each time-step, but offer a handy tool for customising, prototyping, and debugging systems based on RTEC. The translation of Horn clauses to OWL 2 RL is clearly applicable to a number of other event-processing calculi based on Prolog [13, 18, 8, 10]. This approach may hence well set the agenda in event recognition and processing more broadly.

**Acknowledgments.** This work is funded by the EU FP7 INSIGHT project (318225).

**References**


Standardizing the Social Web: The W3C Social Web Activity

Harry Halpin

World Wide Web Consortium/MIT, 32 Vassar Street Cambridge, MA 02139 USA
hhalpin@w3.org

Abstract. The focus of the Social Activity is on making “social” a first-class citizen of the Open Web Platform by enabling standardized protocols, APIs, and an architecture for standardized communication among Social Web applications. These technologies are crucial for both federated social networking and the success of social business between and within the enterprise.

Keywords: decentralization, social web, RDF

1 Introduction

The focus of the W3C Social Activity is on making “social” a first-class citizen of the Open Web Platform by enabling standardized protocols, APIs, and an architecture for standardized communication among Social Web applications. These technologies are crucial for both federated social networking and social business between and within the enterprise and can be built on top of Linked Data. This work will knit together via interoperable standards a number of industry platforms, including IBM Connections, SAP Jam, Jive, SugarCRM, and grassroots efforts such as IndieWeb.¹

The mission of the Social Web Working Group,² part of the Social Activity,³ is to define the technical protocols, Semantic Web vocabularies, and APIs to facilitate access to social functionality as part of the Open Web Platform. These technologies should allow communication between independent systems, federation (also called “decentralization”) being part of the design. The Working Group is chaired by Tantek Celik (Mozilla), Evan Prodromou (E14N), and Arnaud Le Hors (IBM). Also part of the Social Activity is the Social Interest Group ⁴ focuses on messaging and co-ordination in the larger space. This work will include a use-case document, including “social business” enterprise use-cases, as well as vocabularies. The Interest Group is chaired by Mark Crawford (SAP). More information is available in the charters of Social Interest Group and Social Web Working Group.

¹ http://indiewebcamp.com/
² http://www.w3.org/Social/WG
³ http://www.w3.org/Social/
⁴ http://www.w3.org/Social/IG
2 Context and Vision

The Social Activity has been a goal of many members of W3C for years. The Future of Social Networking Workshop\(^5\) was held in 2009 and attracted significant mobile and academic interest, and led to the creation of the Social Web Incubator Group\(^6\) that produced *Towards a Standards-based, Open, and Privacy-Aware Social Web*.\(^7\) Outcomes of this report included the more open Community Group process, since much social web work was happening outside W3C as the W3C was at the time viewed as too exclusive of grass-roots efforts. This also led to further outreach, with the W3C sponsoring and helping organize the grass-roots Federated Social Web conference in 2011. However, at the time there was still not critical mass of W3C members interested in social.

More and more W3C members are embracing the concept of social standards, thank to the work of the Social Business Community Group, in particular the 2011 Social Business Jam.\(^8\) The Social Standards: The Future of Business workshop (sponsored by IBM and the Open Mobile Alliance)\(^9\) developed the standards and ideas for decentralized social networking around industry use-cases. In particular, after the workshop the OpenSocial Foundation joined the W3C, and submitted (with other groups) the OpenSocial Activity Streams and Embedded Experience API as a Member Submission.\(^10\)

3 Goals

The Social Web Working Group will create Recommendation Track deliverables that standardize a common JSON syntax (possibly JSON-LD)\(^11\) for social data, a client-side API, and a Web protocol for federating social information such as status updates. This should allow Web application developers to embed and facilitate access to social communication on the Web. The client-side API produced by this Working Group should be capable of being deployed in a mobile environment and based on HTML5 and the Open Web Platform.

There are a number of use cases that the work of this Working Group will enable, including but not limited to:

- **User control of personal data:** Some users would like to have autonomous control over their own social data, and share their data selectively across various systems. For an example (based on the IndieWeb initiative), a user could host their own blog and use federated status updates to both push and pull their social information across a number of different social networking sites.

---

\(^5\) [http://www.w3.org/2008/09/msnws/](http://www.w3.org/2008/09/msnws/)
\(^6\) [http://www.w3.org/2005/Incubator/socialweb/](http://www.w3.org/2005/Incubator/socialweb/)
\(^7\) [http://www.w3.org/2005/Incubator/socialweb/XGR-socialweb-20101206/](http://www.w3.org/2005/Incubator/socialweb/XGR-socialweb-20101206/)
\(^8\) [http://www.w3.org/2011/socialbusiness-jam/](http://www.w3.org/2011/socialbusiness-jam/)
\(^9\) [http://www.w3.org/2013/socialweb/](http://www.w3.org/2013/socialweb/)
\(^10\) [https://www.w3.org/Submission/2014/SUBM-osapi-20140314/](https://www.w3.org/Submission/2014/SUBM-osapi-20140314/)
\(^11\) [http://www.w3.org/TR/json-ld/](http://www.w3.org/TR/json-ld/)
– **Cross-Organization Ad-hoc Federation**: If two organizations wish to co-operate jointly on a venture, they currently face the problem of securely interoperating two vastly different systems with different kinds of access control and messaging systems. An interoperable system that is based on the federation of decentralized status updates and private groups can help two organizations communicate in a decentralized manner.

– **Embedded Experiences**: When a user is involved in a social process, often a particular action in a status update may need to cause the triggering of an application. For example, a travel request may need to redirect a user to the company’s travel agent. Rather than re-direct the user to another page using HTTP, this interaction could be securely embedded within page itself.

– **Enterprise Social Business**: In any enterprise, different systems need to communicate with each other about the status of various well-defined business processes without having crucial information lost in e-mail. A system built on the federation of decentralized status updates with semantics can help replace email within an enterprise for crucial business processes.

4 **Scope and Deliverables**

The Working Group, in conjunction with Social Interest Group, will determine the use cases that derive the requirements for the deliverables. Features that are not implemented due to time constraints can be put in a non-normative “roadmap” document for future work. The scope will include:

– **Social Data Syntax**: A JSON based syntax (possibly JSON-LD) to allow the transfer of social information, such as status updates, across differing social systems. One input to this deliverable is ActivityStreams 2.0.\[12\]

– **Social API**: A document that defines a specification for a client-side API that lets developers embed and format third party information such as social status updates inside Web applications. One input to this deliverable is the *OpenSocial 2.5.1 Activity Streams and Embedded Experiences APIs Member Submission*, but re-built on top of Linked Data with more secure Javascript sandboxing.

– **Federation Protocol** A Web protocol to allow the federation of activity-based status updates and other data (such as profile information) between heterogeneous Web-based social systems. Federation should include multiple servers sharing updates within a client-server architecture, and allow decentralized social systems to be built. One possible input to this task is WebMention\[13\] and another possible input is the Linked Data Platform.\[14\]

Each of these technologies should not be tightly-coupled but can allow general purpose use. Each specification must contain a section detailing any known

---

\[12\] http://tools.ietf.org/html/draft-snell-activitystreams-05

\[13\] http://indiewebcamp.com/webmention

\[14\] http://www.w3.org/TR/ldp/
security and privacy implications for implementers, Web authors, and end users. The Social Web WG will actively seek an open security and privacy review for every Recommendation-track deliverable.

5 Conclusion

For the Web to break free of centralized proprietary silos, standards are necessary for a decentralized social web to interoperate. We welcome everyone from enterprise to hackers to join this effort to, as put by Tim Berners-Lee, “re-decentralize” the Web.

6 Acknowledgements

This work is funded in part by the European Commission FP7 European Commission through the DCENT Project, which creates privacy-aware tools and applications for direct democracy and economic empowerment.
Efficient Application of Complex Graph Analytics on Very Large Real World RDF Datasets

Zhe Wu
400 Oracle Pkwy
Redwood Shores, CA 94065
alan.wu@oracle.com

Jay Banerjee
One Oracle Dr.
Nashua, NH 03062
jayanta.banerjee@oracle.com

RDF [1] Graph modeling is a foundational technology in the whole semantic web (SW) technology stack. Since its debut in 2004, RDF graph has enjoyed many applications in the enterprise domain. Examples of these applications include, but certainly not limited to, integration and federated query of heterogeneous data sources, flexible and extensible representation of enterprise knowledge base, ad-hoc query and navigation on top of schema-less graph model of enterprise data, social network representation and link analysis, and metadata processing in the context of master data management (MDM). In the past decade, many mature open source and commercial RDF platforms and solutions [6] have been developed to store and index RDF graph data (triples and quads), edit and manage OWL [2] ontologies, perform logical inference, execute pattern matching and graph navigation (SPARQL [3]), visualize RDF graph data and OWL ontologies, and link data in RDF format and also other data types including relational (RDB2RDF [4, 5]). As a graph modeling language, RDF provides great flexibility for enterprise applications and it adds precision, through the use of URI and formal semantics, to enterprise data. SPARQL query and OWL inference have been two key functions for semantic web applications. A somewhat less obvious application of RDF is that such a graph model is also a great candidate for graph analytics.

Large-scale graph analytics [7-10] (page ranking, community detection, etc) for enterprise applications have many challenges, including efficient and scalable graph data management, high-performance implementation of graph algorithms, user- and operation-friendly management interfaces, and tight integration with high quality tools. To effectively address such challenges, we propose a comprehensive graph analytics architecture built upon the Oracle platform. Key components of this platform include Oracle Database 12c, SQL-based graph analytics, parallel in-memory graph processing engine, and the RDF Semantic Graph capabilities. We show why complex graph-based analytics matter for large enterprise-scale RDF datasets, and we share our experiences in implementing several graph analytical functions, such as page ranking, clustering, path analysis, and so on. We apply and evaluate our implementation on several large-scale real world RDF graph datasets, including
graphs from social networks, social media domain and the linked data domain. We make best practice recommendations based on our experiences.

References

1. http://www.w3.org/TR/rdf11-concepts/
2. http://www.w3.org/TR/owl2-syntax/
3. http://www.w3.org/TR/rdf-sparql-query/
4. http://www.w3.org/TR/rdb-direct-mapping/
5. http://www.w3.org/TR/r2rml/
6. http://www.w3.org/wiki/LargeTripleStores
SKOS as a Key Element in Enterprise Linked Data Strategies

Andreas Blumauer
Semantic Web Company GmbH, Mariahilfer Straße 70/8, 1070 Vienna, Austria
a.blumauer@semantic-web.at

Abstract. The challenges in implementing linked data technologies in enterprises are not limited to technical issues only. Projects like these deal also with organisational hurdles to be crossed, for instance the development of employee skills in the area of knowledge modelling and the implementation of a linked data strategy which foresees a cost-effective and sustainable infrastructure of high-quality and linked knowledge graphs. SKOS is able to play a key role in enterprise linked data strategies due to its relative simplicity in parallel with its ability to be mapped and extended by other controlled vocabularies, ontologies, entity extraction services and linked open data.

1 Introduction

The use of semantic web methodologies and technologies has been perceived to be an appealing solution approach for various issues in enterprise information management and data integration [1][2]. Amongst others, the following application scenarios are typically discussed in the context of enterprise linked data: content enrichment and content augmentation, integrated views on distributed data (enterprise mashups), knowledge visualisation, smart assistants and search-driven applications. In all cases, linked data graphs build the basis for such applications, thus the following question is most central for a linked data strategy: How can an enterprise create and maintain knowledge graphs in a sustainable way, whereas the corresponding processes should be as cost-effective as possible and the resulting graphs should be of quality grades which are acceptable for enterprise information services.

1.1 Creating Knowledge Graphs with the Simple Knowledge Organization System (SKOS)

Since the Simple Knowledge Organization System (SKOS) has become a W3C recommendation in 2009 [3], several scenarios to make use of SKOS ontologies have been described [4][5] and many discussions around key design principles such as "minimal ontological commitment" have been led [6]. The increasing use of SKOS can be documented by two key facts:
1. SKOS concept is among over 108,000 classes the most used RDF class in the Linked Open Data cloud.

2. When NISO has published ISO 25964 – the international standard for thesauri and interoperability with other vocabularies, one of the main efforts was to reach the goal of interoperability with SKOS and other schemas.

The usage of SKOS as a starting point to create knowledge graphs in enterprises has in parallel to its relative simplicity (in contrast to other ontology languages like OWL) one other main advantage: It has been accepted broadly as a standard and is well understood by various stakeholders (database engineers, information professionals, knowledge managers), thus little force is needed to overcome the resistance to the introduction of something new like SKOS based vocabularies.

1.2 SKOS as a nucleus of large enterprise knowledge graphs

The scope of a full-blown enterprise knowledge graph is much broader than a taxonomy would be able to cover. When taking a closer look on it, we will find all kinds of categorized and annotated legacy data and documents, additional schemas which describe various business objects, their specific relations and attributes, and linked data graphs from third-party sources, especially from the linked open data cloud [7]. Additionally, a large knowledge graph will contain a lot of mappings between resources from different (named) graphs. What role can SKOS based graphs play in this complex information system?

When starting with SKOS thesauri to describe all kinds of ‘things’ (or ‘business objects’), their names and relations to each other, we don’t have to think about classes or any kind of restrictions or axioms yet. This makes it easy to build a first robust layer of business semantics on top of distributed and heterogeneous information sources. SKOS is based on RDF, thus an extension by additional schemas (classes and properties) is feasible out-of-the-box, at least from a technical point of view. Either custom schemas or already existing ones like FOAF, ORG or schema.org can be used to put additional semantics on top of a SKOS based knowledge graph.

For example, a video game which has been created initially as a SKOS concept with the preferred label ‘SimCity’ and as a narrower concept of another SKOS concept labeled with ‘Video Game’ will be classified in a second step by http://schema.org/SoftwareApplication which is a subclass of http://schema.org/CreativeWork, whereas both are an RDF class derived from schema.org [8]. By using additional schemas, we can express more specific semantics around SKOS graphs and we can map them more accurately with already existing database schemas.

---

1 http://stats.lod2.eu/rdf_classes?sort=overall (accessed on September 15, 2014)
2 http://www.niso.org/schemas/iso25964/#skos (accessed on September 15, 2014)
In addition, when linking SKOS concepts with resources from linked data graphs like Geonames or DBpedia, we can harvest vast amounts of facts around concepts, e.g. birth dates, number of employees, longitude, latitude, etc.

When asking the principal question, why haven’t we started the modelling process with a more complex schema from the very beginning, we should consider the following two aspects:

- **Domain experts are one of the most valuable resources when creating enterprise knowledge graphs.** They most often have no or little expertise with ontology modelling. Thus, they feel more comfortable with a bottom-up approach which starts with concrete instances of classes, and not with a rather abstract schema.

- **Although the ontology of SKOS offers only a few ways to express semantics explicitly, the implicit semantics of a SKOS thesaurus is rich enough in many cases to be made available explicitly and machine-readable by applying additional ontologies.** One of the most principal design patterns of a linked data architect should be, to convert implicit semantics of existing information sources into explicit semantic models based on standards, not the other way around!

## 2 Integrating Knowledge Graphs in Enterprise Information Systems

Enterprise information systems benefit from using knowledge graphs in different ways. The following three scenarios shall illustrate some options:

1. **Knowledge graphs are browsed by end-users and serve as a knowledge base.** In order to make company knowledge better accessible, interfaces should be integrated in popular platforms like Microsoft SharePoint or Atlassian Confluence. As an example for this scenario serves Semantic Knowledge Base for SharePoint\(^3\) which is based on standards-compliant semantic knowledge graphs providing a user interface seamlessly integrated in SharePoint.

2. **Knowledge graphs are used to link and to index information from various sources.** In many cases it will be used for automatic tagging of enterprise information. A knowledge graph can also be used for concept-based search and to generate more complex queries than usually used by simple full-text search. Examples and online-demos for this approach can be found at the PoolParty Semantic Integrator\(^4\) site.

---

\(^3\) [http://www.semantic-sharepoint.com/?page_id=11](http://www.semantic-sharepoint.com/?page_id=11) (accessed on September 16, 2014)

3. Knowledge graphs can also be used in enterprises to analyse and to visualise complex contexts and correlations between business objects. As part of a linked data strategy, components like these should be standard-compliant to increase the chance of being reused all over the places which increases usability. In this concrete example mentioned above, only a SPARQL-endpoint is required to deploy the user application.

3 Conclusion

Knowledge graphs play a central role when establishing linked data based enterprise information systems. Application scenarios are manifold, but creation, maintenance and the actual use of it can be a tedious process. In this paper we described some strategies to develop linked data infrastructures which have turned out to be practically applicable. The use of SKOS to get started with knowledge graphs is one of the key elements in an enterprise linked data strategy.

References

1. Frischmuth, P., Auer, S., Tramp, S., Unbehauen, J., Holzweißig, K., & Marquardt, C. M. (2013). Towards Linked Data based Enterprise Information Integration. In WaSABi@ISWC.

For example, visit http://vocabulary.semantic-web.at/semweb/2184.visual to find a visual representation of 'Linked Data'
The voice of the customer for Digital Telcos

V. Richard Benjamins  
David Cadenas  
Pedro Alonso  
Telefonica, Spain  

Antonio Valderrabanos  
Josu Gomez  
Bitext, Spain

Abstract
In the midst of the digital revolution, the telecommunications industry is undergoing major changes. One of the changes affecting telcos is the increase in data sources from which to get customer feedback – big data, social media. Where this used to be fully controlled by companies through their call centres, websites and shops, today much feedback is expressed in social media, blogs, news sites, app stores and forums. Telefonica has taken up this challenge and opportunity, and is now systematically listening to the voice of its customers online.

The problem
Only a few years ago (around 2010), telcos received a hard wake-up call when Whatsapp started to significantly decrease SMS revenues. Since then, the choice for over-the-top (OTT) products and services has multiplied by orders of magnitude, and telcos run the risk to loose the end-customer contact, and to be forced into a connectivity-only offering. The answer of the telecoms industry to this threat has many aspects (beyond the scope of this paper), including the launch of digital services such as financial services, security, video, etc; leaner working methodologies (lean start-up); and much more customer-driven development and in-live management.

Listening to, understanding, and acting on customer insights are key for launching, growing or (rapidly) killing customer propositions. In this work, we present our approach to systematically and automatically listen to customers on the Internet as soon as a product has gone live\(^1\).

Our approach is built around three main concepts: (i) crawling the Internet, (ii) concept identification & sentiment analysis, and (iii) visualisation, and is set up in such a way that any person with “advanced excel skills” is able to self-serve the needed dashboards in a matter of hours. The result is a system in production for internal use.

Crawler
We use a commercial crawler of Sysomos (www.sysomos.com), which crawls public data as twitter, blogs, news, media and forums on a continuous basis. The input is any Boolean combination of keywords (and, or, not). Selecting the right search terms is important to avoid inclusion of noise. The output is a set of posts, tweets, and articles containing the specific search terms. Where possible,

\(^1\) Notice that customer insights are also key for the conception of new propositions, but that is outside the scope of this paper.
geolocation is provided. We also incorporate reviews from app stores (e.g. Google Play and iTunes) if appropriate. The amount of data will depend on the keywords used in the search. In previous projects where this technology has been applied, we have collected from hundreds of mentions until hundreds of thousands, depending on the subject.

**Concept detection and sentiment analysis**

We use Bitext (www.bitext.com), a company that focuses on analyzing unstructured data (text). Bitext provides an API that receives as input the set of retrieved “items” and as output provides:

- The concepts mentioned in the items
- A set of possibly multiple opinions that constitute the items. E.g. a tweet or news item may contain several opinions about different concepts
- The neutrality or degree of tonality of the opinions (how positive or negative)
- The concepts the opinions are about
- The phrases used to express the tonality of opinions (sentiment)

Bitext applies semantic and linguistic technology to perform those tasks (further explanations on its technology are available on the Internet, e.g., [http://www.bitext.com/deep-linguistic-analysis-critical-successful-text-analytics.html](http://www.bitext.com/deep-linguistic-analysis-critical-successful-text-analytics.html)). We currently use it for Spanish, English and Portuguese. Sentiment is assigned to opinions based on domain-specific ontologies – including tonality- which we have tuned for “digital products” (e.g. in the world of digital products, “cheap” is usually something positive, whereas in general, it can be both positive and negative).

The granularity of sentiment output, which includes the sentiment concept and sentiment text, makes the tool particularly useful for business analysis. Out of the box, Bitext’s technology is about 70% accurate, and after tuning to the digital domain this is increased to 80%-90% (figures obtained from tests on hand-tagged corpora compared to the result of software processing). Typical size of a hand-tagged corpus is +1000 sentences. We follow this process with every major project, where tagging is done by domain experts. The interpretation of irony is a significant area for improvement.

![Words that express sentiment](image)

*Figure 1 Concept cloud representing how sentiment about objects is expressed. Size represents number of opinions; colour represents tonality as in next figure.*

**Visualisation**

For visualisation, we use Tableau (www.tableau.com), which is an easy-to-use (both for development and for viewing) tool for building interactive dashboards.
Interaction allows users to filter for viewing only negative or positive comments, or for different languages, to drill down into more detail, and to always review the original content. For each product or service monitored, we use the following dashboards: Mentions *(by date, source, location, language, most active users & influence factor, most shared content)*, locations *(geography of mentions – the crawler provides about 60% of the mentions)*, concept clouds, sentiment *(concepts triggering opinions, phrases expressing sentiment, degree of tonality, date)*, app store review analysis. Figure 1 shows a concept cloud of how tonality is expressed, and Figure 2 shows the breakdown of tonality of the opinions.

![Figure 1 Concept Cloud of Tonalities](image)

**Conclusions & learnings**

We have learned that –apart from the typical reputation tracking that social media analytics is used for- it is a valuable tool for getting quick and economic customer feedback and insights for products. The tool is able to detect specific issues people complain about, such as for example customer care quality, price and pricing issues, registration process, and specific product features like crashes, unclear interfaces, battery drain, etc.

One thing business users see as very positive is the fact that the tool is available from day 1 of launch, which enables quick responses to typical overlooked product issues, and complements the internally available product KPIs such as downloads, registrations, active users, etc. In general, we see that in the early days after commercial launch, comments are mostly positive reflecting the fact that most are announcements and promises of the great features of the product. Over time, more and more feedback comes in based on actual usage of the product.

Regarding the 70%-90% accuracy of the sentiment analysis software of Bitext, we have learned that when there are thousands of mentions per month, this does not cause a major problem. The dashboards mostly show aggregated information and the main trends, concerns, issues, etc. come out clearly. However, when there are less than 100 mentions a month, especially false positives (e.g. something seen as negative which in fact is not or vice versa), harm the credibility of the tool towards business users.

A final learning is that for some business owners, it is not easy to deal with a lot of negative feedback. And the fact that it is so easy to get feedback and that it is based on publically available knowledge makes it harder to “hide” the insights.
This is however above all a cultural issue. In the lean, digital world, negative feedback should be embraced and taken as an opportunity to quickly improve products based on real customer insights.

In this brief paper, we discussed the opportunity for telcos (and in general, large enterprises) to take advantage of social media analytics as a valuable and economic tool for obtaining customer insights. This is however just one step in the journey to become a full digital telco, which eagerly listens to any relevant external and internal data source about customer and markets, including internal product data, call centre data, Open Data from government initiatives, paid-for-data, analyst data, screen-scraped data, and APIs.
Health and Environment Monitoring Service for Solitary Seniors

Kwangsoo Kim¹, Eunju Lee¹, Soonhyun Kwon¹, Dong-Hwan Park¹, and Seong-il Jin²

¹IoT Convergence Research Department, ETRI, Republic of Korea
{enoch,leenej,kwonshzang,dhpark}@etri.re.kr
²Department of Computer Engineering, Chungnam National University, Republic of Korea
sijin@cnu.ac.kr (Corresponding Author)

Abstract. In this paper, we propose a health and environment monitoring service for solitary seniors using semantic web technologies (ontology modeling, semantic annotation, and ontology-based context awareness). This service defines and logically deducts the environmental conditions which threaten the health of the solitary seniors. This service includes solitary senior, guardian, social workers, health sensors, environment sensors, and energy sensors. By sensing the health status and the indoor environment and extracting events from them, this system provides more efficient welfare services for the solitary seniors.

Keywords: semantic context · healthcare · solitary senior · sensor

1 Introduction

We live in an aging society. According to reports [1, 2, 3], Korea became an “aging society” in 2000 when the number of seniors aged 65 or older reached 7.2% of its total population. In 2018, Korea will become an “aged society” when the seniors comprise 14.4% of its population, and a “super-aged society” by 2027 when the seniors comprise 21.8%. The rapid changes in demography have brought two challenges, care and support, to seniors. Traditionally, Korean seniors have depended on their adult children for care and financial support. In 2011, the number of solitary seniors who live alone in a home comprise 19.6% of Korean seniors, senior couples in which only the couple of old people live 48.5%, seniors living with their children 27.3%, and others 4.6%. Specifically, the number of solitary seniors is increasing rapidly. A lot of them suffer from poverty, disease, and loneliness because their family network is broken. Recently, a solitary senior was found two weeks later after died.

The Korean government enacted Elderly Welfare Act in 1981 in order to contribute to the promotion of the welfare and stability of life of the elderly. Furthermore, August 3, 2007, the Korean government added the contents associated with support for solitary seniors to the act. According to 2 of Article 27 of the act, the state or local governments shall provide both protective measures such as a safety check and services such as visiting home care for solitary seniors. Recently, a congressman proposed an amendment of the act for supporting solitary seniors including 1) emergency medical support and medical treatment visits, regular health check; 2) safety accident prevention in winter and fire; 3) vaccination of infectious diseases; 4) support for emergency food winter; and 5) part or all of the auxiliary cooling and heating costs. According to the act, Korean government has made various policies for the care of solitary seniors. For example, a senior welfare center remotely monitors the movements of solitary seniors using a passive infrared ray (PIR) sensor installed in their house. An automatic monitoring system also is installed in the houses of solitary seniors. An energy utility has developed a system to check remotely the safety of solitary seniors using the amount of power consumption. These methods can efficiently check the safety of solitary seniors in certain ways. However, these methods do not consider the health of solitary seniors as well as the indoor environment affecting the health of them. Therefore, we have developed a system to provide more efficient welfare services for the solitary seniors.

2 Service Description

The left figure of Fig. 1 shows the architecture of the caring system. The system consists of a caring plug, COMUS platform, a caring server, and an application. A caring plug is installed in a house of a solitary senior and measures the indoor environment. The plug includes temperature, humidity, light, PIR, and noise sensor. The PIR sensor reports the number of times the senior has moved during a day. The noise sensor reports the noise level. The plug automatically registers itself in COMUS platform when it wakes up. COMUS platform
which we implemented as Internet of Things (IoT) platform manages all sensor nodes and includes semantic functionalities such as ontology, semantic analyzer, semantic repository, SPARQL engine, linked open data, and semantic translator. A caring server includes the personal information on solitary seniors and social workers, and the business process rule to provide a welfare service. An application installed in a smartphone of a social worker is used to send the health status of a senior to the server and receive an emergency message from.

The right figure of Fig. 1 shows the ontology representing the relations among objects. The ontology consists of a solitary senior, a social worker, a guardian, health sensors, environment sensors, and energy sensor. A solitary senior, a social worker, and a guardian are inherited from foaf:Person object. A social worker in the system is a person who is appointed officially by a particular institution (e.g., a senior welfare center) to take care of a solitary senior and a guardian is selected from relatives of a solitary senior. A solitary senior has environment sensors and energy sensors. A social worker has health sensors. Health sensors measure the health status of a solitary senior, which includes blood pressure, pulse, thermometer, and blood sugar. Environment sensors measure the indoor environment of the house where a solitary senior lives in. They are temperature, humidity, light, PIR, and noise. The energy sensor measures the amount of power which an appliance consumes. Those sensors are inherited from resource ontology included in COMUS (Common Open SeMantic USN Service Platform) platform and installed in a caring plug.

A semantic annotation [4] is performed by the semantic translator using a translation rule and the caring ontology model. To do this, we define a translation rule to perform the semantic annotation. First, we analyze the type of semantic annotation based on ontology vocabularies. Table 1 shows the type of semantic annotation [5]. In Table 1, r indicates a target ontology namespace, rdf indicates a rdf name space, and owl indicates an owl namespace. The type of a semantic annotation is used to define the translation rule for understanding the structure of creating RDF triple pattern. The translation rule defines the method of mapping each elements of the RDF triple pattern into the target ontology model or the value and type of the literal.

<table>
<thead>
<tr>
<th>TYPE</th>
<th>FORMAT (S P O)</th>
<th>EXAMPLE (S P O)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 1</td>
<td>owl:instance rdfs:type owl:Class</td>
<td>ra_23 rdfs:type r:SensorNode</td>
</tr>
<tr>
<td>Type 2</td>
<td>owl:instance owl:ObjectProperty owl:instance</td>
<td>ra_23:produces rob_100</td>
</tr>
<tr>
<td>Type 3</td>
<td>owl:instance owl:DatatypeProperty literal</td>
<td>rob_100:hasValue 10^4^xsd:integer</td>
</tr>
</tbody>
</table>

A semantic processor in COMUS platform extracts events and context from sensing values. An event is extracted from individual sensing value and a context is extracted from one or more events. Specifically, an emergency context with the highest priority is sent to a social worker and a guardian of a solitary senior as fast as possible. For example, an event is that the indoor temperature is too low; a context extracted from events is that a senior is at risk to be frostbite. Fig. 2 shows several user interfaces of the application for a social worker. Fig. 2 (a) is the log-in window which is displayed when a social worker executes the application. The application requires both the user identifier and the password. Then, the worker sees the event history shown in Fig. 2 (b) containing them which were arrived at the smart phone, but have not been checked. Each event is related to the health and the indoor environmental conditions of a solitary senior. The history represents the date and time when the event occurs, senior name, and the description of the event. Fig. 2 (c) shows the list of seniors whom the social worker cares. Fig. 2 (d) shows a senior’s personal data including name, sex, resident registration number, cellular phone number, phone number, address, and other matter. Fig. 2 (e) shows a
senior’s event history including the date and time when an event occurs and the description of the event. Fig. 2 (f) shows a senior’s indoor environmental conditions including temperature, light level, humidity, CO, power usage, movement indicator, and noise indicator which are sensed automatically by the caring plug we installed. By using the values, we will detect whether a solitary senior maintains the daily life or not. Fig. 2 (g) shows a senior’s health data record including the date and time when a health condition was measured, blood pressure, blood sugar, and specific matter. A social worker measures the health condition of a solitary senior and inputs the measurements into the caring system using the interface shown in Fig. 2 (h) including blood pressure, blood sugar, weight, height, and other matter. As the importance of the weight and height of a senior is lower than that of others, they are not shown in Fig. 2 (g).

Fig. 2. User Interface of Caring Service

The proposed service combines the health status and the indoor environment to monitor the daily-life of solitary seniors who do not receive care from their family. By extracting an emergency context and notifying it to a social worker and a guardian in real time, the caring service can increase the safety of solitary seniors. As the proposed system has been operating independently, we need to study the cooperation with several agencies that provide elderly care services in order to provide more systematic elderly care services.

Acknowledgements.
This work was supported by the Industrial Strategic Technology Development Program funded by the Ministry of Science, ICT and Future Planning (MSIP, Korea) [Project No. 10038653, Development of Semantic based Open USN Service Platform].

References.
3. The Korea Institute for Health and Social Affairs: Survey of Elderly (2011)
Abstract. In today's (big) data-intensive world, scalable technologies enabling the efficient management, storage and analysis of large data set are needed. However, the underlying logic of the emerging data-driven business is very different to the established understanding of the traditional often technology-driven industries. As large and complex data are generate almost everywhere in exponentially growth, it is becoming challenging to process and analyze them efficiently by utilizing traditional data analytic and mining techniques. In this paper, we describe the motivation and current needs for semantic web technologies to industry data, where eligible technologies and data storage possibilities are analyzed.

1 Introduction

Semantic web technologies and data mining techniques for unified information access and predictive analytics bring together a multidisciplinary skill set that allows and supports the combination of actual and expected values to plan, predict, and monitor business scenarios and their impact throughout an organization. These techniques play nowadays a key role for challenges such as the optimization of complex system behavior, real-time decision support in operational processes, condition monitoring for predictive maintenance such as failures and fatigue detection, and to increase the efficiency of remote monitoring operations. Especially the processing of data in diagnostics and search related purposes as for instance in alarm management systems become more and more complicated, which can be attributed to the following constraints:

Volume: The diagnosis process, the search for root causes or the calculation of key performance indicators relies on handling large amounts of data. Nowadays, collected data sums up to hundreds of TB for individual use cases [1].

Velocity: In addition to the large amounts of data, more and more data is generated every day. Archived and/or continuous incoming live/streaming data have to be included into the diagnose process to achieve proper results [2].

Variety: Different vendors of machines or single components, coupled with historical or compatibility reasons, lead to multiple different logical and physical data representations and forms. Providing a unified and efficient access to all the different logical models is complex and cumbersome [9].

Veracity: Finally, the aspect of data quality - faulty or missing information leads to high expenses for companies for several reasons. Bad decisions based on wrong information may lead to accidents, resulting in machine damage or even human harm. Additional costs are generated when internal employees are unable to find their required knowledge in time or at all. Consequently, expensive external experts are required [3].

Due to the steady development of new key technologies within the area of semantic web and standards like SPARQL Protocol and RDF Query Language (SPARQL) or Web Ontology Language
(OWL), new approaches and promising ideas emerge to solve diagnosis and search problems also in the area of energy diagnostics. As for instance, automating and offering a general applicable natural language interface [1] and/or ontology-based interpretation [5] reduces the error-proneness and simplifies the query optimization, therefore speeding up the response time. Hence, reducing this amount of time will lead to great benefits for the engineers and companies itself. In this paper, we describe two aspects of a business-driven use case derived from the domain of Energy diagnostics that builds heavily upon semantic web technologies. We describe the benefit of separating traditional and ontology-based data modeling of associated large-scale diagnostic sensor data within a real-time processing setup, and analyze the performance [10] of using RDBMS, RDF and Triple Stores for the knowledge representation.

2 Smart Data Access via Semantic Contextualization

The diagnosis process and searching for root causes within the Energy-based diagnostic context relies primarily on handling large amounts of data. Archived or continuous incoming live data have to be included into the diagnosis to achieve proper results. Nowadays, collected data sums up to hundreds of terabytes. Engineers in the oil and gas industry spend about 30% to 70% of their time searching for data and assessing the quality of the data [4]. As one of the core benefits in which semantic technologies can contribute to this paradigm, is the semantic enhancement and contextualization of traditional processed data by means of ontological information. More precisely, the traditional data acquisition process within the industry context (see Figure 1) is primarily based on the integration of various customer database representations (i.e. Oracle) onto a unified knowledge base. Thereupon, the diagnostic information access is expressed and translated into a SQL-based query language or pattern to derive information about the actual client data and/or to compute the necessary key-performance-indicators (KPI), such as to identify an abnormal behavior of a certain sensor or event (i.e. Show the reliability of appliance A and appliance B? [10]). KPIs such as the reliability concept need thereby to be computed on the frontend-based contextualization/processing engine. That is various complex, and most importantly user-specific, SQL scripts and pattern need to be engineered and customized on client side. In order to formulate and perform the search and analysis task, engineers require additional support from IT Experts more frequently. In contrast to the latter paradigm, the Smart Data Access concept (see Figure 1) enables us to perform these kinds of contextualization (i.e. KPI computation) on and below the actual data acquisition layer. We call this **Configuration Push-Down** and **Analytical Push-Down**. The **Configuration Push-Down** allows to inject the ontological knowledge representation of the industry data prior the actual unified data access layer. That is, it allows to define the concept and the relationships of the comprised client data within an abstract semantic layer - separating data- and knowledge representation. In addition, it supports to define the KPI measures (i.e. availability or reliability) on a concept rather than instance level. That is, KPI measurements are accessible to the user for all data instances of the given concept type. Newly needed KPI definitions do not have a direct impact on the actual client data representation (i.e. changing DB schemas), but can be pushed down my means of replacing and/or enhancing the given Ontology-based knowledge representation. The **Analytical Push-Down** allows an further contextualization by means of analytical computations on the basis of ontological concepts. More precisely, additional events, such as average temperature of sensor X, which are not captured by the traditional data representation, can be integrated as additional context concepts within the domain ontology. In addition, it allows to push-down generic i.e. R functions\(^1\) that are written on a concept level (i.e. \(\text{sensorDiff} = \text{sensorActual} - \text{sensorOptimal}\)), and are executed/computed on an instance level within the data acquisition step. Finally, existing and newly generated information items are

\(^1\) http://www.r-project.org/
stored and accessible utilizing SPARQL on top of an RDF-based triple store. One of the most challenging steps within this data acquisition pipeline is the actual duplication of the client data. Especially within the domain of Energy diagnostics, client sensor data sums up to hundreds of TB. Traditional, client data resides within various (Oracle) relational database systems, and require a well designed (and complete) database layout [10]. Accessing these kind of data in a smart data paradigm, implies the duplication of the data into an RDF-based/supported data access platform, since RDF-based Triple Stores and SPARQL are build to handle and query graph-oriented/ontological data representations. Within the industry context, virtual RDF graphs like D2RQ\(^2\) are to strike a balance between data duplication and semantic contextualization. These systems enable to access relational databases as virtual, read-only RDF graphs. That is, offering RDF-based access to the content of relational databases without having to replicate it into an RDF store. It allows to translate SPARQL queries into SQL queries and return results from a RDBMS in RDF format. However, questions arise on the actual performance and scalability under industrial conditions. In prior experiments ([10]), we have performed a first benchmark analysis, comparing PostgreSQL-based RDBMS, three different native RDF, and virtual Triple Store solution by means of D2RQ, in an industrial data setup. We executed a list of prototype KPI-oriented queries (i.e. Show all reoccurring errors with number of occurrences X within time interval T?) and measured the response time for each of the data stores. The results (see Table 1) indicate that the virtual D2RQ is an interesting approach to access industry-related data, however not competitive in contrast to native and/or RDBMS systems. More precisely, D2RQ had a relatively high but stable query response, reaching up to 30 minutes, which makes it highly impractical for industry-related applications. On the contrary, the native triple stores show a stable fit with respect to their performance time. Sesame performs best with regard to the chosen queries. Overall, virtual stores would be the perfect technology for semantic store integration, since business data does not need to be duplicated, but not yet applicable within industrial perspectives. Native triple stores are mature enough to compete with a RDBMS and still scale well on larger data sets. Moreover, native RDF stores are perfectly suited for the paradigm of Smart Data Access.

\(^{2}\) http://d2rq.org/
<table>
<thead>
<tr>
<th>QueryID</th>
<th>PostgreSQL</th>
<th>Jena TDB</th>
<th>Virtuoso</th>
<th>Sesame</th>
<th>D2RQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>528 ms</td>
<td>4732 ms</td>
<td>4960 ms</td>
<td>4389 ms</td>
<td>8m43s</td>
</tr>
<tr>
<td>2</td>
<td>4290 ms</td>
<td>8561 ms</td>
<td>8801 ms</td>
<td>7837 ms</td>
<td>28m11s</td>
</tr>
<tr>
<td>3</td>
<td>745 ms</td>
<td>6015 ms</td>
<td>5832 ms</td>
<td>5628 ms</td>
<td>10m29s</td>
</tr>
<tr>
<td>4</td>
<td>253 ms</td>
<td>2799 ms</td>
<td>2783 ms</td>
<td>2506 ms</td>
<td>16788 ms</td>
</tr>
<tr>
<td>5</td>
<td>14 ms</td>
<td>115 ms</td>
<td>127 ms</td>
<td>98 ms</td>
<td>207 ms</td>
</tr>
<tr>
<td>6</td>
<td>8023 ms</td>
<td>10512 ms</td>
<td>11172 ms</td>
<td>9848 ms</td>
<td>33m25s</td>
</tr>
<tr>
<td>7</td>
<td>7707 ms</td>
<td>10350 ms</td>
<td>10709 ms</td>
<td>10026 ms</td>
<td>32m56s</td>
</tr>
<tr>
<td>8</td>
<td>7943 ms</td>
<td>10378 ms</td>
<td>10931 ms</td>
<td>10105 ms</td>
<td>31m59s</td>
</tr>
</tbody>
</table>

Table 1. Query response time of the RDBMS PostgreSQL, various native Triple Stores and the virtual Triple Store D2RQ with a 400 MB data conducted on an Intel Q6600 with 4x2400GHz and 16 GB RAM (Table from [10]).

3 Conclusions

In this paper, we described the motivation and current needs for semantic web technologies to industry data by means of contrasting traditional and smart data access. We introduced the concepts of Configuration- and Analytical Push-Down within the Smart Data Acquisition process. We analyze the performance of using RDBMS, RDF and virtual triple stores for the knowledge representation. The results show that virtual RDF graphs do support the DB-driven knowledge graph representation process, but do not perform efficient under industry conditions in terms of performance and scalability.

References

ReApp Store – a Semantic AppStore for Applications in the Robotics Domain

Ana Sasa Bastinos¹, Peter Haase¹, Georg Heppner², Stefan Zander², Nadia Ahmed²

¹fluid Operations, Altrottstraße 31, 69190 Walldorf, Germany
(ana.sasa, peter.haase)@fluidops.com
²FZI Research Center for Information Technology, Haid-und-Neu-Straße 10-14, 76131 Karlsruhe, Germany
(heppner, zander, ahmed)@fzi.de

Abstract. In order to enable a wider use of robot-based automation solutions in medium-sized companies, the purpose of our work is to establish a common repository of robotics applications to facilitate their reuse. This paper presents a semantic AppStore with robotic applications (apps) that does not only contain an app catalog, but also provides a way to find the application one needs by assisting users to find the right application for their purpose, especially if they do not possess the specialized knowledge to decide what software is best suited for the task. The semantic repository of robotic apps is developed in the scope of the ReApp project, hence its name ReApp Store. It is implemented based on real-world requirements of the end-user partners of the project consortium with the purpose to offer real-world apps from the domain of robotics.

1 Introduction

Many companies have a high demand for flexible and economical automation solutions. As the development of such is time-consuming and requires specialized knowledge this usually demands significant investments. The more or less isolated development also prevents sharing and reuse of applications. Due to this, using robot-based automation solutions is often not sensible for medium-sized companies. In order to enable a wider use of robot-based automation solutions in medium-sized companies, the purpose of our work is to establish a common repository of robotics applications. This semantic repository of robotic apps is developed in the scope of the ReApp project [3], hence its name ReApp Store. The ReApp Store is implemented based on real-world requirements of the end-user partners of the project consortium with the purpose to offer real-world apps from the domain of robotics.

2 Users and Purpose of the ReApp Store

The ReApp Store is intended for three types of users: general public, customers of robotics apps, and developers of robotics apps. The main purpose of the ReApp Store is to enable users to find information about the apps that app providers have to offer,
to browse through the apps, to search for the apps, and to download the apps. By fa-
ilitating the semantic properties, the ReApp Store assists users to find the right appli-
cation for their purpose, especially if they do not possess the specialized knowledge to
decide what software is best suited for the task. Furthermore, developers can also
upload their apps to the ReApp Store from their development environment, which
includes semantic descriptions of apps as well app artifacts, such as installation and
support files.

3 Architecture

The ReApp Store is developed as a Linked Data application on the fluidOps Information Workbench platform [1] using W3C standards like OWL, RDF and SPARQL. The Information Workbench is a Web-based open platform for Linked Data and Big Data solutions in the enterprise. In the ReApp Store, we use the OpenRDF Sesame triple store [5] and the Hermit OWL reasoner [4] in the backend, in order to enable semantic wiki pages and to incorporate the ability for semantic querying.

The main architectural aspects of the ReApp Store are:

- **Ontology**: The ReApp ontology is implemented in the OWL ontology language. In the ReApp Store, it is imported into the OpenRDF Sesame triple store. As such it provides a basis for semantic descriptions of apps, presentation of apps, and for implementation of search functionalities.
- **App semantic data**: Semantic descriptions of apps make part of the OpenRDF Sesame triple store. They comply with the ReApp ontology and are used by the Hermit OWL reasoner in order to enable semantic search.
- **API**: Semantic descriptions of apps are imported into the ReApp Store via a standard compliant SPARQL 1.1 endpoint interface provided by the Information Workbench platform. In order to upload app artifacts, a special file upload API is provided.
- **Presentation**: The presentation layer of the ReApp Store is based on the semantic wiki, which makes use of templates and widgets of the Information Workbench in order to present the apps based on their semantic descriptions, and to provide ReApp Store functionalities on the user interface.

4 ReApp Ontology

As the ReApp ontology plays a central role for the presentation of apps as well as for functionality of the ReApp Store, this section presents it in more detail. The ReApp ontology is composed of the base ReApp ontology, the hardware, software and capabilities ontologies. The base ontology defines base classes from the domain of robotic apps, such as types of apps, their properties, messaging used by apps, and relationships between apps and software, hardware and capability concepts.

The hardware ontology contains classification system for different hardware categories based on different output formats generated by hardware components. For
example a CCD Camera is a subclass of a class "ImageSensor" that is a subclass of class “Sensor”. The hardware ontology also categorizes concrete devices. In the ReApp Store, it is used to relate apps with the hardware components they can access.

The software ontology consists of a hierarchy of software categories. It is based on the sensing, planning and acting paradigm [2]. In the ReApp Store, the software ontology is used to relate apps with their software categories. By this relationship, an app can inherit certain category related characteristics and capabilities.

The capability ontology describes capabilities that an app or a hardware component can perform. For example, an app can have a capability “ObjectIdentification” which is a subclass of capability “Perception”.

Moreover, the hardware, software and capability ontologies contain axioms that allow for the computation of capabilities that hardware components and apps exhibit. This enables a reasoner to infer the capabilities needed, e.g. for building a Pick&Place application, or to retrieve a list of components that satisfy a specific capability requirement.

5 Use case

In order to present the real-world deployment of the ReApp Store, we will present a use case of a Pick&Place skill application as it is widely used in industrial packaging applications (Fig. 1). The example Pick&Place skill acts as a server for the action of picking and placing an object for other applications that may use it. It uses a laser scanner and a robot arm with a gripper as hardware components. We will use this skill as an example to demonstrate some of the main advantages of using the ReApp Store.

There are several ways how the user of the ReApp Store could find a Pick&Place skill they were looking for: by performing a semantic search based on the capabilities of this skill (such as planning capability, pick&place capability, etc.), based on the required composite apps, based on hardware devices used, based on ROS interfaces provided or used by this app, or simply browse the catalog of skills. If the user is a developer, they could connect to the ReApp Store from their ReApp development environment and use these types of search from within their environment.

In the ReApp Store, each skill is represented as a composite of several applications. Each component in a skill is represented either as a concrete app or by defining a placeholder. A placeholder is defined as an expression describing certain conditions an app should fulfil. Any app that satisfies the expression can be used at its place. In the ReApp Store, the Pick&Place skill is represented as a composite of several applications, namely two software components and two ROS Wrappers (Fig. 1).

In order to use the sample Pick&Place skill, the user can download the composite apps from the ReApp Store. One of the ROS wrappers in this skill is given as a placeholder that corresponds to the following expression: “ROSWrapper and is used for Depth2DPointCloud and has communication protocol Ethernet.” In order to find apps that can be used in the place of the placeholder, ReApp Store can evaluate the placeholder’s semantic expression and show the list of apps that fulfil the conditions.
6 Conclusion

This paper presented a semantic AppStore of robotic applications. It does not only contain an app catalog, but also comprises semantic descriptions of apps and provides semantic search for apps based on their capabilities and properties. As far as we know, it is the first AppStore of this kind and with it we strive to provide a common basis of apps for the robotics community, i.e. for developers as well as for end-users.

References

Semantic Web based Container Monitoring System for the Transportation Industry

Pinar Gocbe (1), Oguz Dikenelli (1), Umut Kose (2), Juan F. Sequeda (3)
(1) Ege University
(2) Bimar Information Technology Services
(3) Capsenta

Abstract: Goods are transported around the world in containers. Monitoring containers is a complex task. In this paper, a Container Monitoring System based on Semantic Web technologies is presented. This system is currently being developed by Ege University, Bimar Information Technology Services and Capsenta for ARKAS Holding, one of Turkey's leading logistics and transportation companies. The paper consists of 1) introducing the challenges of monitoring containers in the transportation industry, 2) how existing technologies and solutions do not satisfy the needs, 3) why Semantic Web technologies can address the needs, 4) how we are using Semantic Web technologies including architectural design decisions and finally 5) describe lessons learned.

Problem: Monitoring Containers in the Transportation Industry
Logistics and Transportation Industry works as a complex system where different companies interact with each other in a dynamic and adaptive setting. ARKAS Holding is one of Turkey's leading logistics and transportation companies. It operates in different fields such as sea, land, rail, air transportation, ship operations and port operations. In the logistic domain, the main objective is to transport a container from a start location to a final destination. One of the most important problems in this process is monitoring containers at run-time. Each step of the container transportation process may be performed by a different company. Therefore, monitoring the container's lifecycle in real time is a challenging engineering task since these processes runs in parallel on different software systems. The end goal is to have managers and customers be able to track the container transportation life cycle.

Why Semantic Web technologies?
In the last decade, Electronic Data Interchange (EDI) based standards (EDIFACT, RosettaNet, STEP, AnsiX12), XML standards and Service Oriented Architecture (SOA) approaches are used for solving the integration problems of logistics and transportation industry [1]. The standards provide common syntax for data representation and SOA provides an application integration infrastructure. However, these technologies are not sufficient to ultimately solve the integration challenges in large enterprise for the following reasons:

- In EDI-based standards, EDI messages are pushed among the organizations on a predefined time and these standards are not suitable for real-time applications [2].
- In the SOA approaches, the most important problem is interoperability[3]. Unique identifiers in a database, are understood inside of a system but they may lose their meaning in another system. Finding the operation of a web service that will be called by an identifier is fixed in the software application logic which makes application logic hard to understand and vulnerable to the changes,

In the recent years, Semantic Web standards and infrastructure are prevalently used to integrate enterprise information and business processes [4]. Semantic Web provides an integration environment which is more flexible, extensible and open to the exterior when necessary. The same identifiers (URIs) can be used across different data sources creating then a huge knowledge base. Software systems can use this knowledge base independently from each other. Semantic Web technologies decreases the dependence on the middle-tier technologies whose management is hard; so maintenance and management processes are become easier. For all these reasons, Semantic Web technologies comprises a new solution to the dynamic, distributed and complex nature of the logistics and transportation industry.
In ARKAS, there are approximately 200 active integration projects being carried out domestically and internationally. Development cost of a new integration between operational systems to supply tracking data each other is ~25-30% and maintenance cost is ~10-15% of the total project cost. The integration of a new database is performed in approximately one month because of the data format identification and transformation process between the different technologies that are being used.

**The Project**

Ege University, Bimar Information Technology Services and Capsenta are working together to develop a container monitoring system based on Semantic Web technologies\(^1\). The goal of using Semantic Web technologies is to decrease the cost of integration and dependencies between software systems. The first phase consists of integrating four internal systems of ARKAS which are used to manage the processes of ports, agencies, land transport and warehouses. The second phase consists of integrating external relational databases from third-party companies such as a warehouse and land transportation companies.

**How we use Semantic Web technologies**

In order to address the problem, we use a hybrid architecture consisting of a combination of Extract-Transform-Load (ETL), Wrapper, Warehouse and Federation. We have created OWL ontologies that describe port, agency and warehouse processes and R2RML mappings between the relational databases and the ontologies. ARKAS\(^1\) internal databases are ETLed to RDF using Capsenta’s Ultrawrap and warehoused in an RDF triplestore. The goal of having a centralized RDF triplestore is to have full control of the data in a single repository and perform analytics over the integrated data. External data sources are integrated into the system in a distributed manner. External relational databases are wrapped with Ultrawrap in order to provide a virtual RDF view. A query federator is used to integrate the external sources with the internal sources. A wrapper is ideal for the external sources because third-party companies are not willing to give up their data. In order to keep updates to the underlying relational databases consistent with the RDF data in the triplestore, we use data capture systems for relational databases.

**Project Current Status**

Project’s first phase aims to integrate ARKAS’s internal systems using ETL approach. The implementation of first phase has been completed and the team are now working with users for the deployment and validation of the developed Container Monitoring System. This phase has been implemented in an iterative and incremental style. In the first iteration, Agency and Port Management Systems were integrated and tested. Then Warehouse System and Land Transport System were added in second and third iterations. Each iteration finished with a Container Monitor Application which works on the integrated ETL data of the current iteration and this application was used and validated with ARKAS’s domain experts. Each iteration was developed by implementing three consecutive tasks: “Creating Ontology”, “Linked Data Construction”, “Linked Data Application Development”

1. **Creating Ontology**
   As a first task, we created ontology(ies) of the data source(s) which are subject to the current iteration. We used OntoClean [5] and NEON [6] methodologies in order to define structure and semantics of the ontologies. At the current stage, Agency, Port, Warehouse and Transport ontologies have been developed and validated by competency questions defined by domain experts. Also a Core Logistic ontology was defined which contains the core concepts identified during ontology development.

2. **Linked Data Construction**

---

\(^1\) This project is funded by the Republic of Turkey Ministry of Science, Industry and Technology
In the Linked Data Construction step, we created R2RML mappings and converted iteration data sources to RDF format by using Ultragrap. Then, published linked data sources were queried by using competency questions to verify data correctness. Finally, RDF data sources were ETLed and stored in a RDF store. Consequently, we are starting to implement an ETL + Wrapper mode architecture for the second phase of the project. The developed Semantic ETL architecture is depicted in Figure 1. In container monitoring system, approximately one million transactions per day is realized. Our requirement is to handle this transaction volume in real time. For this reason, we used a high performance changed data capture tool (Oracle Golden Gate) to handle changes that occur in the relational databases. It sends each change to a java message queue (JSM) using a pre-defined XML format. We also used highly scalable event driven agent system (Akka Actors\(^2\)) to handle message queue. Each Update Actor is an AKKA agent which handles xml messages in the JMS and converts them to the RDF triple. Concurrency, parallelism and fault tolerance are managed by AKKA infrastructure. Finally, RDF store is updated according to obtained RDF triples.

![Diagram](image)

**Figure 1. Integration Module.**

3. **Linked Data Application Development**

Container Monitoring application has been developed at the end of each iteration. Currently, application integrates all internal data sources and provides a tree and grid style user interfaces to track containers and/or bookings. Figure 2. shows the lifecycle of a specific container

\(^2\) Akka: [http://akka.io/](http://akka.io/)
(UESU5016250) in a tree style. When user clicks Port (anchor symbol) or Agency symbols, container's transactions are listed and details are represented if user clicks any transaction. Monitoring application currently consists of ~220 million triples and it is growing day by day. External customers of the ARKAS and brokers of the ARKAS use the application to track containers and/or bookings. In addition, monitoring application has revealed the importance of the integrated data and a new project has been initiated to use this data for analytical purposes.

![Image](image.png)

**Figure 2. User interface.**

**Current Lessons Learned**
- Given the complexity of the transportation domain, creating ontologies for this domain is not straightforward. Existing ontologies do not satisfy our use case.
- Creation of R2RML mappings involves a domain expert and an ontology engineer. For example, it took 15 days to create the mappings for the port database.
- Deciding on the appropriate linked data architecture according to the requirements is a complex task.
- Creating a simple core ontology and mapping it with suitable R2RML patterns are important tasks to provide a scalable architecture.
- Soft requirements like scalability and/or performance should be evaluated at the end of each iteration and decisions about the software architecture should be taken at the beginning of each iteration based on the evaluation.

**References**