



Linked Data for Transaction Based Enterprise Interoperability

Erwin Folmer, Dennis Krukkert

► To cite this version:

Erwin Folmer, Dennis Krukkert. Linked Data for Transaction Based Enterprise Interoperability. 6th International IFIP Working Conference on Enterprise Interoperability (IWEI), May 2015, Nîmes, France. pp.113-125, 10.1007/978-3-662-47157-9_11 . hal-01438395

HAL Id: hal-01438395

<https://inria.hal.science/hal-01438395>

Submitted on 17 Jan 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Linked Data for transaction based Enterprise Interoperability

Erwin Folmer¹, Dennis Krukkert²

¹Univertity of Twente, Enschede, Netherlands

e.j.a.folmer@utwente.nl

²TNO, Soesterberg, Netherlands

dennis.krukkert@tno.nl

Abstract. Interoperability is of major importance in B2B environments. Starting with EDI in the '80s, currently interoperability relies heavily on XML-based standards. Although having great impact, still issues remain to be solved for improving B2B interoperability. These issues include lack of dynamics, cost of implementations, adoption and cross-industry exchange. Linked Data (part of the Semantic Web) technology, although originally not intended for the B2B domain, holds the promise of overcoming some of these issues.

This paper explores the potential of linked data technology within a B2B context by introducing and studying six scenarios for combining from light to heavy weight 'traditional' standards with Linked Data technology.

This research shows that using Linked Data technology has most potential for specifying semantics formally. This provides the 'best of both worlds' solution, in which legacy systems remain unaltered, and developers are supported in (semi) automated generation of transformation schema's to overcome different standards.

Keywords: Semantic Web, Linked Data, standards, standardization, interoperability.

1 Introduction

Achieving interoperability in many industries is challenging but has great impact. Studies of the US automobile sector, for example, estimate that insufficient interoperability in the supply chain adds at least one billion dollars to operating costs, of which 86% is attributable to data exchange problems [1]. Later studies mention 5 billion dollars for the US automotive industry and 3.9 billion dollars for the electro technical industry, both representing an impressive 1.2% of the value of shipments in each industry [2]. The adoption of standards to improve interoperability in the automotive, aerospace, shipbuilding and other sectors could save billions [3].

The already huge importance of standards and interoperability will continue to grow. Networked business models are becoming indisputable reality in today's economy [4]. A recent Capgemini study concludes that to be ready for 2020 companies

need to “significantly increase their degree of collaboration as well as their networking capability” [5].

Standards are important for ensuring interoperability [6]. “Standards are necessary both for integration and for interoperability” [7]. “Adopting standards-based integration solutions is the most promising way to reduce the long-term costs of integration and facilitate a flexible infrastructure” [8]. Some go even further: “Inter-organizational collaboration requires systems interoperability which is not possible in the absence of common standards” [9].

In an almost completely separated world, new developments take place under the umbrella of Semantic Web, Linked Data and even Big Data. Applications in the business transactions domain however are scarce. The question arises whether these two words can be combined.

2 Research Approach

In this paper is explored if, and how concepts from the Linked Data world, can be used in a different area: the world of inter-organizational interoperability where standardized message exchange for transactions is current practice but has some limitations.

In this explorative research a multi method approach is used. First of all requirements to the solution space are gathered. The requirements are related to the current problems in the area of inter-organizational interoperability: the solution needs to solve identified problems otherwise it seems pointless. The authors are experienced in developing standards message based solutions for many industries and therefore have knowledge about the current limitations. Second, key assets from the Linked Data world are identified through literature search in the key journals (such as Semantic Web journal). Both these experts’ based problems, and the outcome of the literature search are presented in the Background section.

Subsequently, scenarios are identified on how linked data can be used. These scenarios are structures using a very common structure for decomposition of transactions. The scenarios are tested, and validated in a workshop with linked data experts, and iteratively the scenarios are sharpened and pros and cons are gathered. Finally conclusion are presented in the final section.

3 Background

This section presents a background on the inter-organizational interoperability issues and continues with an exploration of Linked Data as background for defining potential solutions.

3.1 Interorganizational Interoperability

Business transaction standards reside at the presentation and application layer of the OSI model [10]. They include semantic standards, inter-organizational information system (IOS) standards, data standards, ontologies, vocabularies, messaging standards, document-based, e-business, horizontal (cross-industry) and vertical industry standards. Examples are RosettaNet (electro technical industry), HealthLevel7 (health care) HR-XML (human resources industry) UBL (procurement). Semantic standards are designed to promote communication and coordination among organizations; these standards may address product identification, data definitions, business document layout, and/or business process sequences (adapted from [10]).

EDI and XML transaction based message exchange for enterprise interoperability have led to tremendous impact in the B2B world [11, 12]. However, still not all domains use the potential. Also, not all interoperability issues are solved, and new issues are introduced [13, 14]. Below we summarize the current issues:

Adoption issues: Many, both XML and EDI based, standards are not being used, or at least less than expected, leading to lower network effects and benefits.

Dynamic issues: The business world is changing, requiring flexibility from standards. This flexibility exists within standard for covering unforeseen business needs and variations of data or business processes, but is not harmonized. Also, many new versions for a standard arise, lowering interoperability.

Implementation cost issues: complexity of standards often lead to costly implementation projects. A part of these costs re-occur for every new version.

Quality issues: standards often offer different implementation choices for the same issue (relates to Dynamic Issue), and loads of optional elements. Different choices lead to interoperability issues. Also, semantics of the elements of the standard, data dictionary and associated rules are not always interpreted in the same way.

Limited interoperability in practice issue: Recent work shows that even a highly successful standard with acclaimed positive benefits does not necessary lead to interoperability on technical/syntax level. This might be caused by a conceptual mismatch: Business people do not want plug and play e-business, 80% interoperability might be enough [15].

Conceptual issues: standards often prescribe, or at least put restrictions on business processes. Although not proven, but still often heard that standards then limit innovation. For example: an innovation in business process will lead to a new version of the standard. However, the restrictions are needed since our conceptual goal has been set on automated business processes: plug and play e-business. Also, our economy and legislation traditionally is based on the notion of (paper) transactions. However, transaction often include information that has been exchanged before, and information that is not always needed. This transaction based thinking therefor has major impact in the message exchange.

Cross sector issue: Many of the current standards are developed for a single sector. Also standards exist that cover functional domain (such as procurement or invoicing). In the networked economy sectors become intertwined, introducing the issue of multiple, not interoperable standards.

Technology issue: There is currently still a lot of old technology in place, caused by the success of EDI/XML based standards. Migrating to newer technology has no positive business case simply because the “old situation” is working more than sufficient for many industries.

3.2 Linked Data (Semantic Web)

“The Semantic Web is here to stay” [16]. The Semantic Web is a vision by Tim Berners-Lee expressed in 2001, about the Internet evolving from a web of documents into a web of data (Web 3.0). Web 3.0 extends current Web 2.0 applications using Semantic Web technologies and graph-based (open) data [17]. In practice and literature terms like Semantic Web, Linked Data or Web 3.0 are used reversibly [18], and although its existence is way before the introduction of the rage around big data, linked data has become uncontroversial part of the Big Data Landscape [19].

The Semantic Web introduces fundamental paradigm shifts such as ‘Anybody can say Anything about Any’: The AAA principle, that can be extended to AAAAA if space and time are being added [19]. Which in practice means that multiple views (truths) can exist regarding a certain dataset. Another paradigm shift is that data should be kept at the source, without exchanging or duplicating the data, but referring (linking) to the source. So, information exchange contains references (URIs) to the source.

Hitzler and Van Harmelen [20] introduce the viewpoint that “semantics is a (possibly unobtainable) gold standard for shared inference” and based on that raises the questions: Why would a shared set of inferences have to consist of conclusions that are held to be either completely true or completely false? This questions the everlasting idea that all information being exchanged has to be complete and valid. In practice it means that not all necessary information for the task at hand will be made mandatory for sender to exchange, but only the information at hand will be exchanged regardless if that is enough for the task at hand for the receiving party. Although an interesting thought, but will it hold for a high value transaction data related to invoicing, or ordering products in the enterprise transaction context?

The goal of defining data semantics as well as the ideal of having a clear formal representation of semantics has not changed, but changing is the way of capturing and using data semantics as well as the formalisms for representation [21]. Data semantics can be used for semantic search, but also for data integration purposes: it is widely acclaimed that ontologies can play a valuable role for semantic data integration by providing a unified structure for linking information from different sources by providing a common interpretation of terminology used in different sources. On the same level it has been shown that semantic models are important for linking ontologies and schemas to each other. Typical use of semantic models is dis-ambiguation of terms, to derive implicit semantic relationships between data items and for detecting inconsistencies that arise due to wrong matches [21]. However many researchers seem to forget that ontologies are not made for their own sake, but that the purpose of an ontology is to help foster semantic interoperability between parties that want to exchange data [20].

Linked data uses RDF (triples) as basic data representation language to vanish syntactic issues, and uses vocabularies that are created in formally well-defined language such as OWL [19]. Triplication is often done without deep contemplation of semantic issues, or of usefulness of the resulting data [20]. A major source of interoperability problems is, however, the different vocabularies and ontologies that are used. And ontology matching in practice is often problematic, partly because semantic heterogeneities tend to be more subtle and owl:sameAs is not sufficient and misleading in practice, [22] and often rather abuse [20].

Again linked data brings in a paradigm shift: from resolving heterogeneity to accounting for it and acknowledging the importance of local conceptualizations by focussing on negotiation and semantic translation [22]. In this regard context becomes an important concept which is largely determined by space and time [22]. Others think that solving ontology interoperability problems is not the right direction, but the aim should be on preventing ontology interoperability problems by developing ontologies on a central (national) level and designing a system of mutually aligned domain ontologies [23].

Several researchers emphasize the distinction between modelling and encoding, with an emphasize on encoding (over modelling) within the Semantic Web community [24, 25]. Modeling semantics is a design task, encoding it is an implementation [24].

Although Semantic Web intended for web data, the technology is much broader useful. For instance Verma & Kass [26] describe usage in requirements engineering, functional and technical design for software engineering. Semantic Web is about semantic interoperability, which is also seen as important layer within inter-organizational interoperability. Semantic Web is about offering support for complex information services by combining information sources that have been designed in a concurrent and distributed manner [25], a situation similar to the domain of inter-organizational information exchange.

4 The Scenarios for Linked Data applications.

There are different options to use Linked Data for enterprise interoperability. In this chapter, first the different scenarios are identified. Then, in section 4.2, the different scenarios are compared and the most potential scenario is identified.

4.1 Identification of the different scenarios

In identifying the different scenarios, variation was applied in two aspects in which linked data and tradition (enterprise interoperability) standards differ.

First aspect is the exchange paradigm. Traditional standards rely on exchanging messages at times that a relevant event has occurred. For example: a product has been sent. In linked data however, the paradigm is not exchanging information, but keeping data at its source, link them to one another, and query for information once its needed.

The second aspect is the way information is expressed and specified. In traditional standards XML messages are exchanged that are (more or less) digital representatives of paper messages that were used before (e.g. an invoice). The structure (syntax) of message instances is expressed in a separate schema. The semantics of information exchanged is typically expressed in a document written in natural language, and thus not interpretable for machines. In linked data, everything is expressed as triples, and there is no strict separation between instances and specification of these instances. Also, semantics can be expressed more formally.

Combining variation in both aspects led to seven possible scenarios which are described below.

State of practice. XML messages are exchanged which are based on XML schema. The schema specifies the structure (syntax) of the message. Typically a (PDF) document is written that provides, in natural language and sometimes UML models, the definition of each of the elements in the schema. Very often this document still gives too much space for interpretation, and therefore a (national or sectoral) localization is written in addition, or as replacement, of the natural language document. **Fig. 1** illustrates information exchange as we know it today.

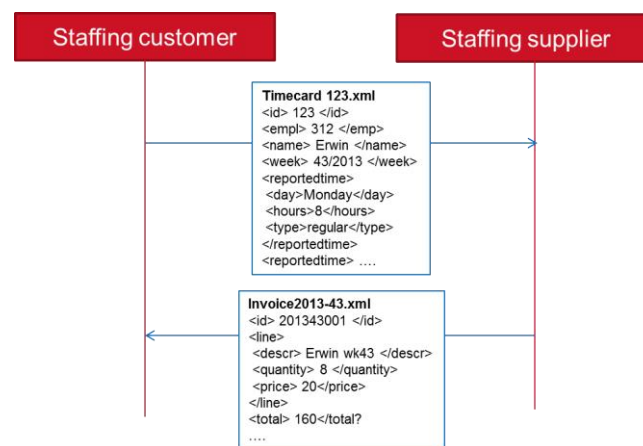


Fig. 1. Traditional exchange of XML messages

“All in” semantic web. In this case no documents are exchanged. Instances, definitions and semantics are expressed using semantic web technology. Each organization has its own triple store to store information, and companies link to one another. For example: a timecard is stored as triples in the triplestore of the customer, while the invoice is stored as triples in the triplestore of the supplier. The invoice only references the timecard. **Fig. 2** illustrates this.

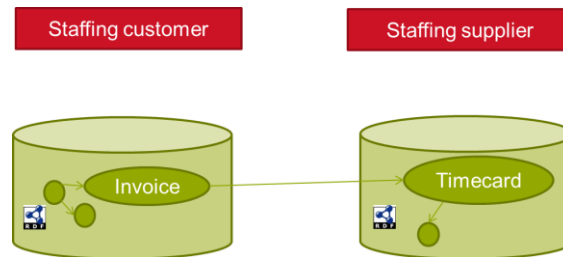


Fig. 2. All in semantic web

At first sight this option might seem like the ultimate B2B solution. Semantics are made explicit and interpretable by machines. Also, one doesn't get any closer to the best practice of 'keeping data at its source'. But, there are some serious hurdles that have to be taken.

The first big issue is, maybe in contrast to what one might have expected, not a technical but a legislative issue. Current legislation is based on the notion that, in order to do business, organizations exchange 'business document'. For example Dutch tax authority states that "An invoice is a document that contains". In other words: exchanging documents is 'part of our system'. The implications are huge.

The promise of making semantics explicit and machine interpretable also needs a side note. Although it is possible to make explicit that one concept is the 'sameAs' another concept with another name (and a computer can reason with this), there are limitations to the expressiveness of semantic web technology. In practice it will still be necessary to have a document that, in natural language –and therefore not interpretable- specifies the different concepts.

A more 'technology driven' issue, or at least an issue that might be solved by technology, is the loss of notification. Traditionally, receiving a 'document' triggers an event. When using semantic web technology, there is no 'receiving'. So how to trigger an event? There are some initiative working providing a solution to this problem, but not is commonly used.

A more fundamental problem is driven by the 'open world' assumption behind the semantic web in combination with the lack of specifying structure. This means that there is no 'semantic web' counterpart of a 'mandatory field'. To take the Dutch authority example again: an invoice must contain a unique number. In XML schema this can be enforced so invoice instances without a unique number will not validate. In the semantic web this will not happen. A reasoner will just assume that such a number exists 'somewhere', even if this is not the case.

Security is also a major concern: you don't want your competitors to have insight in your data. SparQL endpoint do not contain any security or access policies. In practice this is mostly solved by putting a webservice in between that acts like an api.

A final issue is that the installed base of enterprise software will not support semantic web technology, making the introduction very difficult.

Semantic web based on messaging paradigm. Some of the issues mentioned in the previous scenario can be solved by actually exchanging the information besides storing it in a (local) triple store. RDF offers the possibility of serializing different triples in a XML file that in its turn can be exchanged with other organizations. From within the XML serialization, references can be made to the original triples stored in the triple store. **Fig. 3** gives an example.

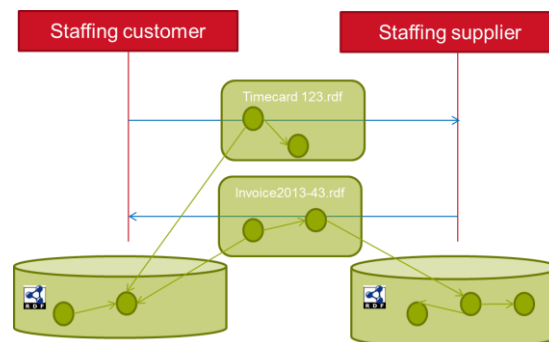


Fig. 3. RDF with an exchange paradigm

By exchanging the information in a RDF/XML serialization, one could argue that the issue of not exchanging messages is dealt with. Also, the issue of notifications is being dealt with. In return, some of the main advantages (like keeping data only at the source) are scarified. Also, the remaining issues mentioned in the “all in” semantic web scenario are not solved.

XML messages based on semantic web ontology. In the “state of practice” scenario several issues are mention when adopting semantic web technology for B2B transactions. Even if this is solved, the installed base of enterprise software nowadays is accommodated to exchanging XML messages, not using to RDF based ontologies.

So, in order to accommodate current enterprise software solutions, it would be better to stick with exchanging XML messages. This scenario investigates the possibilities to actually do this, using a Linked Data ontology for the message definition, instead of XML schema.

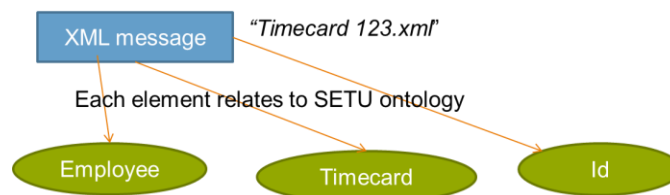


Fig. 4. XML message based on ontology

the drawback of this approach is that the possibility to check whether a XML instance complies to a standard (typically done using XML schema) is lost. Also, these ontologies, typically expressed in RDF, lack the expressiveness of XML schema when it comes to specifying structure.

Current messages and schema, based on ontologies for semantics. The previous section already mentions that for legal and legacy reasons, it is preferred to keep on using XML messages for the information exchange. Also in this scenario, XML messages will still be exchanged, however these XML messages will be based on tradition XML schema. This way structural conformance and completeness can be checked.

The XML schema is linked to an ontology. All concepts (elements) from the XML schema will be expressed as objects in a linked data ontology. Doing so, semantics can be described in a more precise, but moreover: machine interpretable, way.

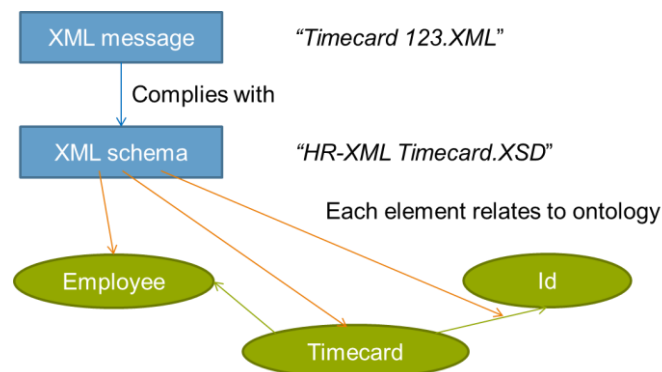


Fig. 5. XML schema linked to ontology

The big advantage of this approach is that it is fully compliant with current (legacy) implementations. As a matter of fact, current implementations don't have to be changed since they will keep on using XML messages and XML schema.

So what advantage does the linked data ontology then provide us? The main difference with current standards is that semantics are expressed in a machine interpretable way. Also, concepts originating from different standards can be 'linked' to one another. Although a couple of practical and more fundamental problems will be encountered, as will describe below, this approach does offer a starting point for automated transformations between different standards.

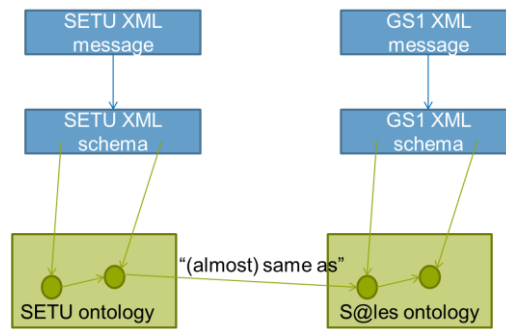


Fig. 6. Mapping between SETU and S@les

As shown in **Fig. 6**, a mapping can be defined (at the ontology level) between elements in different XML schema's. This mapping might be useable for automatic generation of XML transformation schema's (XSLT). Even though this systematics has already been implemented in a prototype (prestoprime.joanneum.at) to transform between different media metadata formats, we still see some hurdles to be taken for B2B applications.

One issue is that some elements in a schema are used for 'structuring purposes', and are not actual 'real world' concepts. Typically this is done for 'containers'. An example from the SETU standard is 'reference information' which contains various elements that can be used for referring to other objects or documents.

If two elements in different XML schema's have a different syntax, but semantically the same, then transformation is rather straightforward. However, a more fundamental problem is that in most cases elements from different standards are (semantically) not exactly the same. In order to make transformation possible in such a case, it is needed to explicitly express what the differences are. Current technology isn't capable to do this in a sufficient way.

Although the more fundamental problem stated above prevents automatic generation of transformationsheets (XSLT), it could support the designer a great deal by giving suggestions on what elements are potentially the same. If there are two standards that need a transformation scheme this doesn't offer a lot of gain. On the other hand, if there are tens of standard (e.g. electronic invoice standard), then it would help the designers a lot.

Mixed content: codelists based on RDF. Semantic standards typically contain a lot of codelists. There are a different ways to use codelists in a standards: a table in the PDF document of the standard, reference to an external PDF document, enumeration as part of the schema, and finally: importing an external schema. Every option has it's advantages and disadvantaged.

When a codelist is in a PDF document (either as part of the standard or externally referenced), it is not possible to do automatic validation. When a codelist is part of the standard (either in the PDF document, or as enumeration in the schema), the dynamics are very low since the codelist can only change with the standard. Also, when a external codelist is used as source, manual synchronization is needed.

Linked data on the other hand is by nature very well suited to be 'maintained elsewhere'. So, one can imagine a situation where schema's 'link' to a codelist that is maintained elsewhere. This does however require changes to legacy software to cope with this kind of codelists.

Combined content: RDF as additional info in a standard message. All previous scenario's choices were made at different levels of the interoperability stack, for either 'the traditional' way or the 'linked data way'. There is however also another option: combine both.

The most obvious way to do this is by using RDFa to add concept from a Linked data ontology into a traditional XML message. Every element can be accompanied by a RDF counterpart. This does however require a change to the schema's. But, once realized, one can choose to add (optional) RDF data to a message.

The combination of traditional XML messages and RDFa seems a nice approach for a 'transition period', but it would require additional effort from IT systems sending the messages. This also raises questions: which parties would be interested in putting effort in creating messages that contain the same data in two formats, while the parties that receives the data will only support one.

4.2 Analysis of the scenarios

In the "All in" semantic web scenario we already mentioned that there is a fundamental issue on how to express that two concepts are 'more or less the same'. Moreover: how to explicitly and precisely express what the difference is between two 'more or less the same' concepts is. Also, since 'anyone can say everything about anything': how does one know how reliable such a statement is, and how does one know that the context in which such a statement is made also suits the context in which the statement is going to be used. For example: a staffing company might conclude that a Human Resource is more or less the same as a Person. An university might state that a Student is more or less the same as a Person. Giving this info the question arises what conclusions can be drawn. For most people a student and a Human Resource have a lot in common (both natural persons). However, for a procurement officer a Human Resource is more or less the same as a box of nails (both can be 'purchased').

We think the "Current messages and schema, based on ontologies for semantics" scenario is the most potential: linking a XML schema to an ontology, and use that ontology to help with creating transformation schema's.

Using an ontology for (semi) automated definition of transformation schemes can be implemented in two ways. The first option is to have an intermediate solution (during 'runtime' exchanging of messages) that receives XML messages in one format, does the transformation, and forwards the message in another format. For doing the transformation, the intermediate solution directly accesses and uses the knowledge in the ontology.

The second option is to first (in 'design time') distill an XML transformation schema (XSLT) based on the two original XML schema's and the ontology, and use this XSLT when exchanging messages.

The advantage of the first approach is that one is not limited to the expressiveness of XSLT (although we're not sure if this poses a problem), while the second approach had the advantage that a lot of the enterprise services busses that are used today support XSLT.

5 Conclusions

Linked Data (Semantic Web) is an important technology approach within the container concept of Big Data. It is being developed to transform the document-centric world of the Internet transforming into a web of data instead of documents. However, the technology looks promising for the business transaction (e-business) world as well, although it was never designed with this application in mind.

This business transaction world has a long history of interoperability challenges covered by many standards based solutions starting from EDI solutions in the '80s to XML based standards that are used a lot nowadays. These solutions made an enormous positive impact but still several issues remains unsolved. This includes issues in the area of adoption, dynamics/flexibility, high implementation cost, quality, cross sectoral exchange and legacy solutions.

This paper aims to answer the question if Linked Data can contribute to solving these issues. Linked Data contains both conceptual and technical aspects. Eg. The principle that data is kept on the source and not being copied, just as the adagio: Anybody can say anything about anything, are examples on the conceptual level. Owl, RDF (the object-subject-predicate) and Sparql are examples of technical concepts of Linked Data.

Linked Data holds the promise to solve cross sector interoperability, its ability to handle (slightly) different semantics in communication, reduce redundant information exchange by linking, handles different versioning, make better reuse of existing data.

Six scenario's for inclusion of Linked Data concepts in the Enterprise transaction world are identified. These scenario's range from a full-blown Linked Data scenario down to using a small set of Linked Data concepts. The scenarios can by the way be implemented in incremental steps, making introduction easier.

Although all of the scenario's show a lot of potential advantaged, there are also some serious hurdles to take. One example is that Linked Data isn't meant for expressing structure which means that, combined with the open world assumption of Linked Data, it's very hard to enforce that specific information is actually exchanged. Also, from a legal perspective, the idea that 'exchange of messages' will be lost is a complex one. Other examples are mention in the paper as well.

The most realistic scenario is using Linked data at a 'design time' so support engineers, but at 'runtime' stick to current technology. This means that current XML messages, based on XML schema, will remain to be exchanged. For supporting the engineer, the schema will be related to an 'upper' ontology. For cross-sectoral exchange, the ontology and reasoners will give suggestions on what elements from different standards are potentially the same. Also, Linked Data could be used for specifying and reusing (elements within) codelists.

To sum up, although Linked Data is rapidly gaining importance and practical implementations are more and more common, it doesn't seem realistic that Linked Data within the world of business transactions becomes common in the near future (1-5 years). This paper shows that, although there is much potential in Linked Data, and at glance it seems that Linked Data is easy to implement in the business transactions world, the devil is in the details. And these details are quite essential, especially for

the conceptual ones. However, since there is a lot of potential in Linked Data for business transaction, we urge to do more research on this topic and then aim for some large scale implementations to show the huge economic impact.

References

1. Brunnermeier, S.B. and S.A. Martin, Interoperability costs in the US automotive supply chain. *Supply Chain Management*. 7(2): p. 71-82 (2002)
2. Steinfield, C.W., M.L. Markus, and R.T. Wigand. Cooperative Advantage and Vertical Information System Standards: An Automotive Supply Chain Case Study. in 44th Hawaii International Conference on System Sciences (HICSS). Hawaii (2011)
3. Gallaher, M.P., A.C. O'Conner, and T. Phelps, Economic Impact Assessment of the International Standard for the Exchange of Product Model Data (STEP) in Transportation Equipment Industries, (2002).
4. Legner, C. and B. Lebreton, Preface to the Focus Theme Section: 'Business Interoperability' Business Interoperability Research: Present Achievements and Upcoming Challenges. *Electronic Markets*. 17(3): p. 176-186 (2007)
5. Falge, C., B. Otto, and H. Österle. Data Quality Requirements of Collaborative Business Processes. in 45th Hawaii International Conference on System Sciences (HICSS). Hawaii (2012)
6. Rada, R., Standards: the language for success. *Communications of the ACM*. 36(12): p. 17-23 (1993)
7. Dogac, A., et al., Collaborative business process support in eHealth: Integrating IHE profiles through ebXML business process specification language. *IEEE Transactions on Information Technology in Biomedicine*. 12(6): p. 754-762 (2008)
8. Chari, K. and S. Seshadri, Demystifying integration. *Communications of the ACM*. 47(7): p. 58-63 (2004)
9. Gerst, M., R. Bunduchi, and R. Williams. Social shaping & standardization: A case study from auto industry. in 38th Hawaii International Conference on System Sciences (HICSS). Hawaii (2005)
10. Steinfield, C.W., et al., Promoting e-business through vertical IS standards: lessons from the US home mortgage industry, in *Standards and Public Policy*, S. Greenstein and V. Stango, Editors., Cambridge University Press: Cambridge. p. 160-207 (2007)
11. Wang, E.T.G. and A. Seidmann, Electronic data interchange: competitive externalities and strategic implementation policies. *Management Science*. 41(3): p. 401-418 (1995)
12. Wigand, R.T., C.W. Steinfield, and M.L. Markus, Information technology standards choices and industry structure outcomes: The case of the U.S. home mortgage industry. *Journal of Management Information Systems*. 22(2): p. 165-191 (2005)

13. Damsgaard, J. and D. Truex, Binary trading relations and the limits of EDI standards: The Procrustean bed of standards. *European Journal of Information Systems*. 9(3): p. 173-188 (2000)
14. Rukanova, B.D., K. Slooten, van, and R.A. Stegwee, Business Process Requirements, Modeling Technique and Standard: how to Identify Interoperability Gaps on a Process Level, in *Interoperability of Enterprise Software and Applications*, D. Konstantas, et al., Editors., Springer-Verlag: London. p. 13-23 (2006)
15. Folmer, E. and H. Wu. Semantic Standards Quality Measured for Achieving Enterprise Interoperability: The Case of the SETU Standard for Flexible Staffing. in *IWEI 2013*. Enschede, The Netherlands: ISTE Wiley (2013)
16. Hitzler, P. and K. Janowicz, Semantic Web - Interoperability, Usability, Applicability. *Semantic Web*. 2010(1): p. 1-2 (2010)
17. Hendler, J., Web 3.0 Emerging. *Computer*. 42(1): p. 111-113 (2009)
18. Heath, T. and C. Bizer, Linked Data - Evolving the Web into a Global Data Space. *Synthesis Lectures on The Semantic Web: Theory and Technology*, ed. J. Hendler and F.v. Harmelen. Vol. 1. Morgan & Claypool (2011)
19. Hitzler, P. and K. Janowicz, Linked Data, Big Data, and the 4th Paradigm. *Semantic Web*. 4(3): p. 233-235 (2013)
20. Hitzler, P. and F.v. Harmelen, A reasonable Semantic Web. *Semantic Web*. 2010(1): p. 39-44 (2010)
21. Stuckenschmidt, H., Data Semantics on the Web. *Journal on Data Semantics*. 2012(1): p. 1-9 (2012)
22. Janowicz, K., The Role of Space and Time For Knowledge Organization on the Semantic Web. *Semantic Web*. 1(1): p. 25-32 (2010)
23. Hyvönen, Preventing ontology interoperability problems instead of solving them. *Semantic Web*. 2010(1): p. 33-37 (2010)
24. Janowicz, K., Modeling vs encoding for the Semantic Web. *Semantic Web*. 2010(1): p. 11-15 (2010)
25. Guizzardi, G., Theoretical Foundations and Engineering Tools for Building Ontologies as Reference Conceptual Models. *Semantic Web*. 1(1): p. 3-10 (2010)
26. Verma, K. and A. Kass, Model-Assisted Software Development: Using a 'semantic bus' to automate steps in the software development process. *Semantic Web*. 1(1): p. 17-24 (2010)