Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo

Tesis Doctoral
-
Programa de Doctorado en Informática

Integración semántica de grandes fuentes de datos heterogéneas

Semantic integration of heterogeneous big data sources

Integración semántica de grandes fontes de datos heteroxenees

Herminio García González

Supervisada por:
Prof. José Emilio Labra Gayo y Prof. Juan Manuel Cueva Lovelle

Diciembre de 2020

Agradecimientos

Esta tesis que aquí se presenta ha sido un trabajo de casi cinco años, un desafío con sus dificultades e incertidumbres pero también con sus alegrías. Y aunque parezca que ha pasado mucho y poco tiempo a la vez, llega el momento de cerrar esta etapa y buscar nuevas aventuras y desafíos. Como cualquier etapa en la vida de una persona hay gente que te influye de una u otra manera, y es por eso que esta sección va dedicada a ellos, a rendirles un merecido homenaje.

En primer lugar me gustaría agradecer a mis directores de tesis: José Emilio Labra Gayo y Juan Manuel Cueva Lovelle, por la oportunidad que me han dado y por su fe (muchas veces ciega) en mi manera de hacer y ver las cosas. A Juan Manuel Cueva Lovelle por su apoyo a lo largo de la tesis y por sus sabios consejos. A José Emilio Labra Gayo por involucrarme desde muy temprano en WESO y permitirme descubrir el gran mundo de la Web Semántica.

A todos los compañeros que han pasado por el laboratorio de tecnologías orientadas a objetos, su compañía, ideas y las discusiones que allí tenían lugar han sido muy enriquecedoras durante esta etapa. En especial me gustaría agradecer a Dani—el otro «Wesito» durante esta etapa—, Cristian y Óscar por nuestras largas charlas, el apoyo que siempre me han brindado y esas partidas de pádel tan necesarias ciertos días.

A todos los compañeros del INRIA Lille Nord Europe por la buena acogida durante mi estancia y por los buenos momentos que me hicieron pasar. *Merci beaucoup pour votre accueil chaleureux et pour les bons moments que nous avons passés.*

A los compañeros del Servicio de Informática y Comunicaciones de la Universidad de Oviedo por su buena acogida—aunque viniera del «lado oscuro»—, su compañerismo y el buen ambiente que generan.

A mi familia por su apoyo, interés y comprensión durante esta etapa. No es un mundo sencillo de entender, pero siempre habéis estado ahí para lo que hiciera falta.

A mi pareja, Elena, por ser mi compañera de viaje durante esta etapa y las que vendrán. No es fácil sacar a la vez dos tesis adelante, pero parece que finalmente lo estamos logrando. Sus consejos, apoyo y escucha diaria me han servido para no derrumbarme y seguir adelante. A mis dos pequeñas «perrinas», Zelda y Nala que aunque no entiendan de qué iba esto, me han hecho olvidarme de los problemas en los malos momentos y valorar las pequeñas cosas del día a día.

Y a otros tantos que seguro que me dejo en el tintero. A todos vosotros, gracias.

Resumen

La integración de datos es el problema de agregar diferentes tipos de datos de manera que puedan ser usados mediante una única interfaz. Esto es un problema central de las ciencias de la computación donde los datos están repartidos en silos de información que dificultan su acceso e integración. La aparición de la Web Semántica supuso un avance en cuanto a las tecnologías que tuvieran como propósito central la interconexión de datos y que facilitaran esta tarea. Por tanto, en esta tesis proponemos un lenguaje para integración de datos, ShExML, que busca facilitar las tareas de integración frente a otras alternativas, y que produce grafos RDF como resultado. Así mismo, las técnicas de validación permiten establecer una serie de atributos deseables en un conjunto de datos (confiabilidad, normalización, estandarización, etc.); por lo cual, proponemos la conversión de esquemas a su alternativa dentro de las tecnologías de la Web Semántica, empezando por la técnica de conversión de XML Schema a Shape Expressions (ShEx) descrita en esta trabajo. Con el fin de probar la utilidad de las tecnologías semánticas y, específicamente, del lenguaje implementado se describen dos trabajos llevados a cabo en los campos del e-Learning y las Humanidades Digitales que intentan implementar estas herramientas dentro de los procesos propios de estas disciplinas con el fin de ofrecer nuevas perspecivas y mejorarlos.

Las evaluaciones hechas en este trabajo demuestran que el uso de ShExML mejora el proceso de integración de datos para los usuarios que se inician en este tipo de actividades frente a otras alternativas. La transformación de esquemas propuesta es viable, logramos transformar los elementos de XML Schema a Shape Expressions y la validación de los conjuntos de datos equivalentes se produce adecuadamente. Sin embargo, se produce una pérdida de semántica en algunas conversiones debido a la diferencia de semántica previa entre los dos lenguajes, hecho que hace que la conversión inversa —de Shape Expressions a XML Schema— no sea siempre posible. La inclusión de contenido adicional extraido de la nube de datos enlazados demuestra mejorar la efectividad didáctica de los alumnos frente a la herramienta propia de un LMS. Por su lado, la utilización de ShExML para la transformación de transcripciones de manuscritos históricos en XML-TEI a RDF confiere a estas transcripciones de una serie de atributos alineados con la estrategia FAIR.

A la luz de los resultados obtenidos proponemos mejoras, nuevas funcionalidades y retos que esta línea de investigación tiene que resolver y afrontar en el futuro. Con este trabajo hemos intentado mejorar la migración de datos en tecnologías no semánticas a tecnologías semánticas, así como explorar su uso en otras disciplinas como modo de aprendizaje, retroalimentación y posterior mejora.

Abstract

Data integration is the problem of integrating different kind of data so that they can be used through a single interface. This is a central problem in computer science where data is spread around information silos that can hinder their access and integration. The emergence of the Semantic Web has supposed an advance on technologies which has a special focus on data interconnection and its facilitation. Thus, in this thesis we propose a data integration langugage, ShExML, which has the aim to facilitate this kind of tasks—in comparison with other alternatives—and that produces RDF as the output. Likewise, validation techniques allow to establish certain desirable attributes in a dataset (e.g.: reliability, normalisation, standardisation, etc.); therefore, we propose the schema conversion to their counterpart alternatives within the Semantic Web, beginning with the XML Schema to Shape Expressions translation described in this work. With the aim to test the utility of semantic technologies and, specifically, the utility of the implemented language we describe two works performed in e-Learning and Digital Humanities fields. These works try to apply these tools in these fields processes in order to offer new perspectives and to improve them.

The evaluation carried out in this work demonstrate that the use of ShExML improves the process of data integration for first-time users in comparison with other alternatives. The proposed schemata transformation has showed to be viable as we are able to transform XML Schema elements to their Shape Expressions counterpart. Moreover, the validation of equivalent datasets is performed correctly. However, there is a loss of semantics in some conversions due to the previous semantic difference between both languages. This factor influences the backwards conversion—from Shape Expressions to XML Schema—which cannot be always possible. The inclusion of additional content extracted from the Linked Open Data Cloud has showed to improve the students' didactic effectiveness in relation to LMS own tool. The use of ShExML for the translation from XML-TEI to RDF of historic manuscript transcriptions confers to these transcriptions with certain FAIR aligned attributes.

In the light of the obtained result we proposed some improvement, new functionalities and challenges that this research topic should solve and face in the future. With this work we have attempted to improve the migration of non-semantic technologies to semantic technologies alongside the exploration of their use in other fields as a way to learn, get feedback and, in the end, improve.

Resume

L'integración de datos ye'l problema d'axuntar diferentes tipos de datos de manera que puean ser usaos por mediu d'una única interfaz. Esti ye un problema central de les ciencies de la computación onde los datos tan dixebraos en silos d'información que compliquen el so accesu ya integración. L'apaición de la Web Semántica supuso un avance nes teunoloxíes que tuvieren como'l so propósitu central l'amiestu de datos y qu'estos trabayos seyan más afayadizos. Asina, nesta tesis proponemos un llinguaxe pa l'integración de datos, ShExML, que busca facilitar les tarees d'integración en comparanza a otres alternatives, y que produz grafos RDF como resultáu. Amás, les téuniques de validación permiten establecer una serie d'atributos deseables en un conxuntu de datos (confiabilidad, normalización, estandarización, etc.); polo tanto, proponemos la tresformación d'esquemes a la so alternativa dientro de les teunoloxíes de la Web Semántica, principiando pela téunica de conversión de XML Schema a Shape Expressions (ShEx) esplicada nesti trabayu. Col fin de probar la utilidá de les teunoloxíes semántiques y del llinguaxe desarrollau describense dos trabayos fechos nos campos del e-Learning y les Humanidáes Dixitales qu'intenten incorporar estes ferramientes dientro de los procesos propios d'estes disciplines col fin d'ofrecer nueves perspeutives y ameyorarlos.

Les evaluaciones feches nesti trabayu desmuestren que'l usu de ShExML ameyora'l procesu d'integración de datos pa los usuarios que principien nesti tipu de llabores en comparanza a otres alternatives. La tresformación d'esquemes propuesta ye viable, llogremos tresformar los elementos de XML Schema a Shape Expressions y la validación de los conxuntos de datos equivalentes produzse correutamente. Sin embargu, dase una pérdida de semántica n'algunes conversiones debíu a la diferencia de semántica previa ente los dos llinguaxes, fechu que fae que la tresformación inversa —de Shape Expressions a XML Schema— nun seya siempre posible. L'amiestu de conteníu adicional estrayíu de la ñube de datos enllazaos demuestra ameyorar la efectividá didáutica de los alumnos frente a la ferramienta propia d'un LMS. Pel so llau, la utilización de ShExML pa la tresformación de trescripciones de manuscritos históricos en XML-TEI a RDF dota a estes trescripciones d'una serie d'atributos alliniaos cola estratexa FAIR.

A la lluz de los resultaos estrayíos proponemos meyoras, nueves carauterístiques y retos qu'esta llínea tien que resolver y afrontar nel futuru. Con esti trabayu intentemos ameyorar la migración de datos en teunoloxíes non semántiques a teunoloxíes semántiques, asina como esplorar el so usu n'otres disciplines a mou d'aprendimientu, retroalimentación y posterior meyora.

Índice general

Índice de cuadros

Índice de figuras

Capítulo 1

Introducción

La integración de datos es el problema de agregar diferentes tipos de datos de manera que puedan ser usados mediante una única interfaz [59]. Este problema se puede subdividir en dos subconjuntos: el intercambio de datos (*data exchange*) [42], por el cuál convertimos un conjunto de datos que siguen un esquema X en otro conjunto de datos que siguen un esquema Z siendo $X \neq Z$; y la propia integración de datos (*data merging*) por la cual se agregan varios conjuntos de datos siguiendo esquemas diferentes (X, Y, Z, etc.) en un conjunto de datos que sigue un único esquema (S). Estos dos subconjuntos están relacionados, y en algunas ocasiones, el intercambio de datos puede ser un subconjunto de la integración de datos —aunque no siempre—. Estos dos subconjuntos son los que veremos identificados en los diferentes capítulos cuando hablemos de las soluciones existentes o la propia solución propuesta, por tanto, es interesante tenerlos en cuenta a lo largo de este documento.

Sin embargo, vale la pena detenerse primero en la importancia del tema que se está tratando. La integración de datos surge como una respuesta a un problema endémico de la ciencias de la computación y es la disparidad de formatos y representaciones de una misma realidad. Dicho de otro modo, cada organización tiende a tener sus datos aislados y sin modo de interconexión e integración con los datos de otras organizaciones, haciendo que, aunque haya mucho conocimiento almacenado, es imposible relacionarlo y acceder a él. Este problema es lo que comúnmente se conoce como el problema de los silos de información.

Con el uso de soluciones de intercambio de datos se podrían integrar datos de una base de datos en otra, salvando el problema de la diferencia de formato y representación. Esta estrategia la vemos reflejada en varios ejemplos: el uso de servicios web como forma de interoperabilidad entre diferentes aplicaciones, el uso de mecanismos de procedimientos remotos, las propias librerías de consulta de base de datos en los diferentes lenguajes de programación, etc. Sin embargo, se puede llegar a producir una pérdida de datos si el esquema con el que queremos integrar no soporta algunos datos de nuestro esquema de origen. Por ejemplo, en los esquemas de bases de datos existen términos como claves primarias, claves externas, claves únicas que al ser cargados a objetos Plain Old Java Object (POJO) o serializados en formatos como XML y JSON pierden dicha semántica. Lo mismo sucede al revés y es que XML favorece mucho la secuencialidad (el orden dentro de los hijos de un mismo padre) y las relaciones arbóreas; sin embargo, esto no es tan fácil de representar en un esquema relacional.

Hoy en día se producen, cada vez, un mayor volumen de datos que se hace difícil de manejar y analizar globalmente [108]. Soluciones en el campo de la Inteligencia Artificial [83], IoT [2], o el Big Data [82], nos proveen con herramientas para poder manejar estos datos, recolectar datos desde sensores o hacer nuevas deducciones y aprendizajes desde ellos. Sin embargo, hacen falta soluciones que permitan integrar y tratar todos esos datos conjuntamente. Es ahí donde se quiere enfocar el esfuerzo de este trabajo. Y es este uno de los retos de nuestro tiempo ya no sólo en el campo de las ciencias de la computación sino también en otros campos como el propio de la investigación donde el concepto de datos FAIR [131] (de las siglas en inglés *Findability, accesibility, interoperability and reusability*) es un valor creciente para los datos derivados de investigaciones y promovido desde la propia Comisión Europea[1].

Dentro de las tecnologías que podrían ser adecuadas para el tratamiento de este problema, la aparición de la Web Semántica [8] supuso una nueva perspectiva para el mismo. La Web Semántica propone el uso de *Internationalized Resources Identifiers* (IRIs, por sus siglas en inglés) únicos que permitan hacer relaciones implícitas entre diferentes conjuntos de datos reusando IRIs ya existentes. Además, una característica importante de *Resource Description Framework* (RDF, de sus siglas en inglés) [27], que es el formato de datos propuesto y defendido desde la comunidad de la Web Semántica, es que es composicional. Esto significa que uno puede directamente unir varios conjuntos de datos sin necesidad de hacer uso de un mecanismo de unión. Estas características hacen a RDF un formato privilegiado para la integración de datos. Un buen ejemplo de esto es Wikidata [130] donde múltiples contribuidores —humanos y robots— transforman datos desde diferentes fuentes y las integran a Wikidata donde pueden ser consultados en un formato único por toda la humanidad.

Así mismo, la integración de datos y el intercambio de datos cuentan con otra vertiente que es igual de importante, la validación de los datos. Mediante la validación de datos se puede asegurar una normalización en los datos y establecer una confianza en que los datos estarán estructurados de la manera indicada. Dicho de otra forma, la validación de datos puede favorecer la limpieza de los datos, su consulta y la estandarización de los conjuntos de datos [47]. Es por esto, que además de poder ofrecer soluciones que permitan integrar los datos, es también recomendable poder hacer una transformación e integración de los esquemas de validación.

Siguiendo con la idea de la Web Semántica como vehículo para la integración de datos, es necesario ver qué posibilidades de validación de datos existen dentro de este terreno. Esto es, de hecho, un campo de investigación relativamente reciente donde dos lenguajes, *Shape Expressions* (ShEx) [105] y *Shapes Constraint Language* (SHACL) [69], son las dos soluciones que se proponen desde la comunidad para hacer frente a este problema. Es por tanto interesante ver cómo se podría convertir desde un esquema de origen a un esquema final siguiendo la idea de la integración de datos y su posterior validación. Es decir, si tenemos un conjunto de datos fiables, normalizados y limpios queremos extender estos atributos a las transformaciones que de ellos hagamos y también

[1] `https://op.europa.eu/en/publication-detail/-/publication/7769a148-f1f6-11e8-9982-01aa75ed71a1`

a sus posibles integraciones con otros conjuntos de datos.

El producto final que se obtiene de un proceso como el descrito anteriormente es lo que se ha venido a llamar recientemente en el campo de la Web Semántica como Grafo de Conocimiento (en inglés, *Knowledge Graph*)[2]. Estos grafos de conocimientos son pequeñas piezas de conocimiento que siguiendo los principios de la Web Semántica, vistos anteriormente, permiten representar un dominio concreto y pueden ser integrados con otros grafos de conocimiento, del mismo dominio u otro diferente, haciendo que tras varias uniones se pueda llegar a grafos de conocimiento de cierta envergadura.

Una vez que existen grafos de conocimiento que cubren un dominio importante, de manera bastante amplia, y siguen los atributos de calidad que hemos visto como producto de los procesos de validación, empiezan a ser un producto que puede ser utilizado con gran potencial por otras aplicaciones o usuarios. Recordemos que la Web Semántica no sólo trata de unir datos y darles un formato común, sino también de facilitar el procesamiento de estos datos por la propias máquinas. De ahí el nombre, hacer de la web sintáctica, pensada para ser leída por humanos, una web semántica que pueda ser leída y procesada por humanos y máquinas. El potencial de este tipo de soluciones está empezando a demostrarse, pero aún hay mucho camino por recorrer y muchos campos donde aplicar estos avances. Sin ir más lejos, en esta propia tesis se va a abordar la utilización de estos grafos de conocimiento en un campo de conocimiento y de gran valor como es el aprendizaje en línea o virtual (en inglés conocido como *e-Learning*). Y es que la capacidad de poder adquirir nuevo conocimiento a través de los recursos existentes en la Web Semántica, de poder adaptar los conocimientos con otros nuevos conocimientos en función del perfil del estudiante, (hipermedia adaptativa [22]) o poder navegar por los recursos al ritmo que el estudiante decida, sin limitación técnica; se antoja como una posibilidad muy prometedora a investigar. Es decir, la posibilidad de poder adoptar estas tecnologías en diferentes casos de uso significaría probar de alguna manera la utilidad de todas estas técnicas y soluciones propuestas.

Así mismo, y siguiendo con la posibilidad que la Web Semántica y la integración de datos puede ofrecer para hacer de los datos de investigación datos que cumplan los estándares FAIR, también se investiga la adopción de este tipo de herramientas en un campo otrora ajeno a las ciencias de la computación como las humanidades. El término Humanidades Digitales [112, 10] viene a definir el fenómeno cada vez más en auge y expansión por el cuál las técnicas creadas e investigadas en el campo de las ciencias de la computación pueden ser aplicadas a las humanidades para acelerar sus procesos de investigación y crear nuevas metodologías que permitan avanzar la investigación en dichos campos. Además, y volviendo a los datos FAIR, la posibilidad de preservación de datos históricos y humanísticos usando métodos computacionales hace que este sea un campo muy interesante donde aplicar las técnicas diseñadas en las ciencias de la computación, la integración de datos y la Web Semántica, usando todas estas tecnologías seríamos capaces de preservar, estandarizar y ofrecer al público en general los materiales y los avances que de estas investigaciones humanísticas se derivan [88].

[2] `https://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html`

1.1 Trabajo relacionado

Aunque en los capítulos de esta tesis se repasan los diferentes trabajos relacionados y estado del arte, se hace conveniente hacer una pequeña introducción a los mismos, de manera que se entienda la motivación de cada uno de los trabajos, en que trabajos e ideas se basan, así como cuáles serían otros trabajos que han abordado problemas similares.

Por tanto, empezando por la integración de datos heterogéneos, usando RDF como formato de destino (ver Capítulo 2 para las razones por las cuales RDF es idóneo para este cometido), podemos identificar RML [37] como la primera propuesta para llevar a cabo este cometido mediante un lenguaje de dominio específico (DSL, por sus siglas en inglés). RML extiende la especificación de R2RML[3] que consideraba la conversión de datos en bases de datos relacionales a RDF. Mediante esta extensión se logra generalizar dicha especificación y poder usar un mismo lenguaje con diferentes formatos, mejorando la mantenibilidad y versatilidad del proceso de integración de datos. Después de esta propuesta vendrían algunas más: xR2RML [89] que fue propuesto originalmente para habilitar transformaciones de bases de datos relacionales y NoSQL; SPARQL-Generate [75] que extiende SPARQL 1.1 para hacer consultas sobre datos heterogéneos, busca ofrecer una sintaxis familiar dentro del mundo de la web semántica y cuya implementación mejora en rendimiento a la implementación de referencia de RML [76]; y YARRRML [65] que ofrece una sintaxis sencilla basada en YAML y que ofrece un traductor de sus reglas a RML. Sin embargo, aunque tanto SPARQL-Generate y YARRRML fueron ideados con el objetivo de que fuera fáciles de usar [31] no se ha llevado a cabo ningún estudio o experimento comparativo que pueda demostrar estos argumentos. Este hecho motiva el trabajo presentado en el Capítulo 2.

Una vez que la transformación de datos es posible nos interesa poder transformar esas reglas de validación que tenían los datos de origen a los datos de destino. Sin embargo, recordemos que la validación de grafos RDF es un tema relativamente reciente. Por tanto, es un terreno todavía sin investigar. Dentro de XML (que es el ecosistema en el que nos centramos en el Capítulo 3) hay algún trabajo previo usando formatos no semánticos, p. ej.: de XML Schema a JSONSchema [97] o entre modelos relaciones y XML Schema [74]. Dentro del terreno de la web semántica se ha explorado la conversión de XML Schema a OWL [43, 109] y a RDF Schema [91]. Sin embargo, el uso de ontologías para la validación de datos plantea una serie de problemas debido al uso de las asunciones de mundo abierto y nombre no único (Open World and Non-Unique Name Assumptions, en inglés) [123]. Otra experiencia usando múltiples formatos de validación es la llevada a cabo en FHIR[4]. Sin embargo, en FHIR se hace uso de un modelo del dominio abstracto que luego es traducido a los diferentes formatos. De esta exposición podemos ver que no hay ningún trabajo que permita traducir las reglas de validación de XML Schema a ShEx o SHACL. Es este el motivo de proponer una serie de mapeos entre estos dos formatos y analizar sus consecuencias (ver Capítulo 3).

La integración de datos, y específicamente, los datos albergados dentro de la nube de datos enlazados puede ser de gran utilidad y aplicabilidad en mu-

[3] `https://www.w3.org/TR/r2rml/`

[4] `https://www.hl7.org/fhir/`

chos campos. Como prueba de concepto, hemos explorado su utilidad dentro del campo de aprendizaje en línea (e-Learning, en inglés). Trabajos previos han explorado también esta posibilidad de diferentes maneras: en [81] se exploran técnicas semánticas de minería de datos para proveer experiencias de aprendizaje en línea personalizadas; por otro lado en [132] se etiquetan semánticamente videos educativos para mejorar su búsqueda. Además, las ontologías se han usando para recomendar materiales en base al progreso de los estudiantes [55] y como medio para definir y estructurar material educativo [120]. Así mismo, la mejora de contenido se ha llevado a cabo usando técnicas de Hipermedia Adaptativa [22] como la creación de lenguajes de adaptación [30, 93, 29] o usando objetos de aprendizaje [100, 58]. Dadas estas experiencias previas hemos querido ver la utilidad de los conocimientos albergados en la Web Semántica como medio de aumentar la efectividad didáctica de los alumnos (ver Capítulo 4).

Otro campo en el que las tecnologías semánticas han tenido una gran acogida es el campo de las Humanidades Digitales. Y es que la Web Semántica ha sido vista como una alternativa para la publicación, reconciliación, estandarización e integración en el campo de las humanidades [88]. Por tanto, hemos querido hacer uso del lenguaje de integración creado en el Capítulo 2 para transformar material histórico a datos enlazados, cogiendo como caso de uso el notariado público en Asturias en el siglo XIII. Debido a diferentes particularidades del caso de uso (ver Capítulo 5) las transcripciones no se pueden automatizar y por tanto han sido hechas manualmente y volcadas en ficheros XML siguiendo el vocabulario TEI. Trabajos previos han llevado a cabo transformaciones similares: en [103] se propone una transformación de XML/TEI a RDF/XML, en [36] se hace una transformación similar pero usando XSLT y en [57] usando XTriples[5]. Por tanto en el Capítulo 5 proponemos el uso de languajes de mapeo de datos heterogéneos [31] para mejorar la velocidad y flexibilidad de las metodologías que necesiten transformar datos.

1.2 Preguntas de investigación generales

Una vez visto el trabajo relacionado, y viendo el estado actual del tema que se quiere tratar —la integración de datos heterogéneos—, se plantean las siguientes preguntas de investigación generales:

- PIG1: ¿Se puede mejorar la usabilidad y facilidad de uso de los lenguajes de integración de datos heterogéneos?

- PIG2: ¿Es posible traducir los esquemas de las fuentes de datos de origen de manera que los datos traducidos puedan estar igualmente validados y normalizados?

- PIG3: ¿Se pueden usar grafos de conocimiento para mejorar el contenido de las plataformas de aprendizaje electrónico, y por tanto el aprendizaje de sus usuarios?

- PIG4: ¿Pueden las herramientas de integración de datos ayudar en las metodologías de traducción e integración de datos de las humanidades digitales?

[5] `https://xtriples.lod.academy/index.html`

Estas preguntas de investigación generales derivan en las contribuciones que se presentan en la siguiente sección. Cada contribución genera una hipótesis y diferentes preguntas de investigación específicas que son presentadas en la Sección 1.4.

1.3 Contribuciones

Siguiendo el planteamiento del apartado anterior, en esta tesis se presentan las siguientes contribuciones:

- Un lenguaje para la integración de datos heterogéneos, mostrando que el diseño del mismo puede mejorar la usabilidad de los usuarios que se enfrentan por primera vez a este tipo de soluciones.

- Un mecanismo para poder hacer una conversión automática de esquemas de XML Schema a Shape Expressions (lenguaje de validación de RDF). Se presenta un prototipo que cubre los casos más básicos y se presenta un estudio teórico de las posibilidades de esta conversión, sus limitaciones y otras características de la misma.

- Una herramienta para el enriquecimiento de textos educativos mediante el uso de NLP, desambiguación de entidades y grafos de conocimiento de la Web Semántica. Mediante una evaluación se demuestra su eficacia didáctica.

- Un proceso por el cual se transforman manuscritos previamente transcritos a XML, usando el vocabulario *Text Encoding Initiative* [6] (TEI, por sus siglas en inglés) a RDF usando la herramienta ShExML creada en esta tesis doctoral. En este trabajo se puede ver la aplicación de la herramienta, los resultados que puede generar, así como también se discuten ciertas limitaciones de la metodología propuesta, futuros pasos que habría que añadir a la metodología con el fin de enriquecer la transformación y como esto aporta nuevas ideas para implementar en ShExML.

1.4 Artículos

Dicho esto, y dado que la tesis se presenta como compendio de publicaciones, este trabajo está dividido en cuatro partes que coinciden con los cuatro artículos de los cuáles se compone la misma.

ShExML: Improving the usability of heterogeneous data mapping languages for first-time users

En este artículo se presenta el lenguaje ShExML diseñado como prototipo para la integración de datos heterogéneos produciendo salidas en RDF. Además, se hace una comparativa entre este lenguaje y otras alternativas existentes mirando su expresividad y diseño. Finalmente se presentan los resultados de un experimento en el que un grupo de estudiantes prueba tres de estos lenguajes

[6] `https://tei-c.org/`

y se analizan estos resultados en conjunto con el estudio del diseño y de la expresividad para llegar a conclusiones sobre la usabilidad de los mismos.

Específicamente la hipótesis de este trabajo es:

≪*Los usuarios primerizos con conocimientos de programación y Web Semántica pueden ver facilitada la tarea de integración de datos usando ShExML frente a otras alternativas.*≫

Consecuentemente, las preguntas de investigación son las siguientes:

- PI1: ¿Está el diseño del lenguaje ShExML mejorando la usabilidad para los usuarios primerizos en comparación con otros lenguajes?
- PI2: Si es cierto, ¿se puede establecer una relación entre el soporte de funcionalidad y la usabilidad para los usuarios primerizos?
- PI3: ¿Qué partes de ShExML —y de otros lenguajes— pueden ser mejoradas para incrementar la usabilidad?

XMLSchema2ShEx: Converting XML validation to RDF validation

En este artículo se analiza la transformación de esquemas en XML Schema a esquemas en ShEx pudiendo sentar las bases para la automatización de esta tarea. En el artículo se estudian los diferentes elementos de XML Schema y cómo estos pueden ser transformados en elementos de ShEx equivalentes. Además, se analiza la posibilidad de pérdida de semántica en la conversión, la equivalencia de los dos esquemas y la conversión inversa del esquema generado.

La hipótesis de este trabajo es:

≪Es posible realizar una conversión automática de XML Schema a Shape Expressions manteniendo la equivalencia entre esquemas≫

Las preguntas de investigación planteadas en este artículo son las siguientes:

- PI1: ¿Qué componentes debe tener una conversión de XML Schema a ShEx?
- PI2: ¿Cómo podemos asegurar que ambos esquemas son equivalentes?
- PI3: ¿Es posible asegurar una conversión inversa en todos los casos?
- PI4: ¿Es posible convertir y validar los esquemas no deterministas?

Enhancing e-Learning content by using Semantic Web technologies

En este artículo se presenta una herramienta que hace uso de las capacidades de procesamiento de lenguaje natural y de desambiguación de entidades de Apache Stanbol[7] para extraer entidades destacables de textos educativos, para posteriormente, enriquecer el propio contenido educativo con información extra sobre los propios contenidos. Esto se ofrece como un *plug-in* de la plataforma educativa Sakai añadiendo tarjetas desplegables a las menciones de entidades significativas. Además, se demuestra la efectividad didáctica de esta solución

[7] `https://stanbol.apache.org/`

por medio de un experimento realizado en un instituto de educación secundaria asturiano.

Por tanto, la hipótesis de este trabajo es:

≪La adición de contenido relevante extraído de grafos de conocimiento de la Web Semántica puede mejorar la efectividad didáctica frente a las soluciones basadas sólo en texto≫

En consecuencia, las preguntas de investigación son las siguientes:

- PI1: ¿La adición de contenido extraído de la Web Semántica puede producir una mejora en la efectividad didáctica?
- PI2: ¿Qué percepción tienen los alumnos de este tipo de herramientas?
- PI3: ¿En qué asignaturas sería más útil incluir este tipo de herramientas?

Converting Asturian Notaries Public deeds to Linked Data using TEI and ShExML

En este artículo se explora la posibilidad de convertir manuscritos de notarios públicos de Asturias entre los siglos XII y XIV que han sido transcritos usando un vocabulario de XML ideado para la codificación de textos TEI. La idea detrás de este artículo es hacer uso de la herramienta ShExML, descrita en el Capítulo 2, para hacer la conversión de estos textos en TEI, poder transformarlos a RDF de una manera casi automática favoreciendo su integración con otro tipo de textos similares, ya publicados en RDF, y pudiendo desambiguar las menciones a entidades propias (p. ej.: personas, lugares, eventos) ya existentes en la *Linked Data Cloud*[8]. Esto haría que estas transcripciones, que son fruto de una investigación histórica, pudieran cumplir los preceptos FAIR de una manera rápida y sencilla, sin necesidad de un gran aprendizaje específico. Así mismo, se presentan las líneas futuras de avance de este proyecto y cómo sería necesario definir una ontología que pudiera abarcar los términos necesarios para poder hacer un estudio diplomático usando estas fuentes y esta conversión.

1.5 Estructura

El resto de este documento se estructura de la siguiente manera: en el Capítulo 2 se presenta el artículo sobre el lenguaje de integración de datos, en el Capítulo 3 se presenta el estudio de conversión de esquemas en XML Schema a esquemas en ShEx, en el Capítulo 4 se presenta el artículo sobre la herramienta para enriquecer textos educativos, el Capítulo 5 describe el proceso de transformación de manuscritos transcritos en TEI a RDF usando ShExML, el Capítulo 6 presenta una discusión conjunta de los resultados de esta tesis; y, finalmente, en el Capítulo 7 se extraen las conclusiones de esta investigación y se dibujan las líneas de trabajo futuro que han surgido fruto de la misma.

[8] `https://lod-cloud.net/`

Chapter 2

ShExML: improving the usability of heterogeneous data mapping languages for first-time users

This article was originally published as:

Herminio García-González, Iovka Boneva, Sławek Staworko, José Emilio Labra-Gayo, and Juan Manuel Cueva Lovelle. ShExML: improving the usability of heterogeneous data mapping languages for first-time users. *PeerJ Computer Science*, 6(e318), 2020

Herminio García-González, José Emilio Labra-Gayo and Juan Manuel Cueva Lovelle were with the Department of Computer Science, University of Oviedo, Oviedo, Asturias, Spain
Iovka Boneva and Sławek Staworko were with the University of Lille, INRIA, Lille, Nord-Pas-de-Calais, France

The journal has the following metrics according to 2019 JCR:

- 2019 Impact Factor: 3.091
- 5 Year Impact Factor: N/A
- Computer Science, Artificial Intelligence: Q2 (50/137)
- Computer Science, Information Systems: Q2 (53/156)
- Computer Science, Theory & Methods: Q1 (24/108)

Abstract

Integration of heterogeneous data sources in a single representation is an active field with many different tools and techniques. In the case of text-based approaches—those that base the definition of the mappings and the integration on a DSL—there is a lack of usability studies. In this work we have conducted a usability experiment ($n = 17$) on three different languages: ShExML (our own language), YARRRML and SPARQL-Generate. Results show that ShExML users tend to perform better than those of YARRRML and SPARQL-Generate. This study sheds light on usability aspects of these languages design and remarks some aspects of improvement.

2.1 Introduction

Data integration is the problem of mapping data from different sources so that they can be used through a single interface [59]. In particular, data exchange is the process of transforming source data to a target data model, so that it can be integrated in existing applications [42]. Modern data exchange solutions require from the user to define a *mapping* from the source data model to the target data model, which is then used by the system to perform the actual data transformation. This process is crucial to many applications nowadays as the number of heterogeneous data sources is growing [108].

Although many technologies have appeared through the years, the emergence of the semantic web [8] offered new perspectives for data integration. The semantic web principle recommends to represent entities through a unique Internationalized Resource Identifier (IRI) which allows creation of implicit links between distinct datasets simply by reusing existing IRIs. Moreover, the Resource Description Framework (RDF), which is the advocated data format for the semantic web, is compositional, meaning that one can simply fuse data sources without the use of a specific merger. These characteristics make RDF a privileged format for data integration, thus a target for data exchange and transformation.

The most notable example of an RDF based data integration system is Wikidata[1] where multiple contributors—humans or robots—transform data from different sources and integrate it to the Wikidata data store. Another example is the `data.bnf.fr`[2] project that exposes in RDF format the catalog of the French National Library (BNF) by interlinking it with other datasets around the world.

Initially, the only way to perform these data transformations was to use *ad-hoc* scripts designed to take one data source and transform it to an RDF output. This supposed the creation of a dedicated script for every new input data source that needed to be converted. Such solutions are slow and costly to develop.

Later on, Domain Specific Language (DSL) approaches emerged which are able to define a translation in a declarative fashion instead of an imperative one. This technique lowers the development time, but a script for every different data source is still needed, which can be a maintenance issue.

[1] `https://www.wikidata.org/`

[2] `https://data.bnf.fr/en/about` for more information on the project

More recent systems allow direct transformation of multiple data sources into a single representation. Some of them provide dedicated DSLs in which a single script defines the multi-source transformation, others provide graphical interfaces. This is an improvement compared to previous techniques as in principle it allows for faster development and improved maintainability [84]. However, the adoption of such systems depends also on their *usability* [61].

With usability in mind we have designed the ShExML [50] language that allows transformation and integration of data from XML and JSON sources in a single RDF output. ShExML uses Shape Expressions (ShEx) [105] for defining the desired structure of the output. ShExML has text based syntax (in contrast to graphical tools) and is intended for users that prefer this kind of representation. Our hypothesis is that for first-time users with some programming and Linked Data background, data integration is performed more easily using ShExML than using one of the existing alternatives. The consequent research questions that we study in the current paper are:

- RQ1: Is ShExML more usable for first-time users over other languages?
- RQ2: If true, can a relation be established between features support and usability for first-time users?
- RQ3: Which parts of ShExML—and of other languages—can be improved to increase usability?

In the case of this work we are going to focus on usability of tools based on a DSL and see how the design of the language can have an effect on usability and associated measures such as: development time, learning curve, etc.

The rest of the paper is structured as follows: Section 2.2 studies the related work, in Section 2.3 the three languages are compared alongside a features comparison between them, in Section 2.4 we describe the methodology followed in the study, in Section 2.5 the results are presented along with their statistical analysis. In Section 2.6 we discuss and interpret the results and in Section 2.7 we draw some conclusions and propose some future lines from this work.

2.2 Background

We first review available tools and systems for generating RDF from different systems for data representation. These can be divided into one-to-one and many-to-one transformations. We also survey existing studies on the effectiveness of heterogeneous data mapping tools.

One to one transformations

Much research work has been done in this topic where conversions and technologies were proposed to transform from a structured format (e.g., XML, JSON, CSV, Databases, etc.) to RDF.

From XML to RDF

In XML ecosystem many conversions and tools have been proposed:

[91] describe their experience with the transformation of RDF to XML (and vice versa) and from XML Schema to RDF Schema. [34] propose a transformation from XML to RDF which is based on an ontology and a mapping document. An approach to convert XML to RDF using XML Schema is reported by [4, 6]. [127] describe how they perform a translation from XML to RDF using a matching between XML Schema and RDF Schema. The same procedure was firstly proved with a matching between DTD and RDF Schema by the same authors in [125]. [18] reports a technique for the transformation between XML and RDF by means of the XSLT technology which is applied to astronomy data. Another approach that uses XSLT attached to schemata definitions is described by [117]. However, use of XSLT for lifting purposes tends to end up in complex and non flexible stylesheets. Thus, [13] present XSPARQL, a framework that enables the transformation between XML and RDF by using XQuery and SPARQL to overcome the drawbacks of using XSLT for these transformations.

From JSON to RDF

Although in the JSON ecosystem there are less proposed conversions and tools, there are some works that should be mentioned.

[95] present a transformation of a RESTful API serving interlinked JSON documents to RDF for sensor data. An RDF production methodology from JSON data tested on the Greek open data repository is presented by [124]. [48] report a tool able to identify JSON metadata, align them with vocabulary and convert it to RDF; in addition, they identify the most appropriate entity type for the JSON objects.

From tabular form to RDF

The importance of CSV (along with its spreadsheet counterparts) has influenced work in this ecosystem:

[41] present a mapping language whose processor is able to convert from tabular data to RDF. A tool for translating spreadsheets to RDF without the assumption of identical vocabulary per row is described by [60]. [45] report a platform to import and lift from spreadsheet to RDF with a human-computer interface. Using SPARQL 1.1 syntax TARQL[3] offers an engine to transform from CSV to RDF. CSVW proposed a W3C Recommendation to define CSV to RDF transformations using a dedicated DSL [122].

From Databases to RDF

Along with the XML ecosystem, relational database transformation to RDF is another field:

[15] present a platform to access relational databases as a virtual RDF store. A mechanism to directly map relational databases to RDF and OWL is described by [113]; this direct mapping produces a OWL ontology which is used as the basis for the mapping to RDF. Triplify [3] allows to publish relational data as Linked Data converting HTTP-URI requests to relational database queries. One of the most relevant proposals is R2RML [28] that became a

[3] `http://tarql.github.io/`

W3C Recommendation in 2012. R2RML offers a standard language to define conversions from relational databases to RDF. In order to offer a more intuitive way to declare mapping from databases to RDF, [119] presented SML which bases its mappings into SQL views and SPARQL construct queries.

More comprehensive reviews of tools and comparisons of tools for the purpose of lifting from relational databases to RDF are presented by [90, 63, 110].

Many to one transformations

Many to one transformations is a recent topic which has evolved to overcome the problem that one to one transformations need a different solution for each format and that subsequently must be maintained.

Source-centric approaches

Source-centric approaches are those that, even giving the possibility of transforming multiple data sources to multiple serialisation formats, they base their transformation mechanism in one to one transformations. This can deliver optimal results—if exported to RDF—due to RDF compositional property. Some of the tools available are: OpenRefine[4] which allows to perform data cleanup and transformation to other formats, DataTank[5] which offers transformation of data by means of a RESTful architecture, Virtuoso Sponger[6] is a middleware component of Virtuoso able to transform from a data input format to another serialisation format, RDFizers[7] employs the Open Semantic Framework to offer hundreds of different format converters to RDF. The Datalift [111] framework also offers the possibility of transforming raw data to semantic interlinked data sources.

Text-based approaches

The use of a mapping language as the way to define all the mappings for various data sources was first introduced by RML [37] which extends R2RML syntax (Turtle based) to cover heterogeneous data sources. With RML implementations it is possible to gather data from: XML, JSON, CSV, Databases and so on; and put them together in the same RDF output. A similar approach was also followed in KR2RML [116] which proposed an alternative interpretation of R2RML rules paired with a source-agnostic processor facilitating data cleaning and transformation. To deal with non-relational databases, [89] presented xR2RML language which extends R2RML and RML specifications. Then, SPARQL-Generate [75] was proposed which extends SPARQL syntax to serve as a mapping language for heterogeneous data. This solution has the advantage of using a very well-known syntax in the semantic web community and that its implementation is more efficient than RML main one (i.e., RMLMapper[8]) [76]. To offer a simpler solution for users of text-based approaches, YARRRML [65]

[4] http://openrefine.org/

[5] http://thedatatank.com/

[6] http://vos.openlinksw.com/owiki/wiki/VOS/VirtSponger

[7] http://wiki.opensemanticframework.org/index.php/RDFizers

[8] https://github.com/RMLio/RML-Mapper

was introduced which offers a YAML based syntax and its processor [9] performs a translation to RML rules.

Graphical-based approaches

Graphical tools offer an easier way to interact with the mapping engine and are more accessible to non-expert users. Some of the tools mentioned in the previous source-centric approaches section have graphical interfaces, like OpenRefine and DataTank. RMLEditor [64] offers a graphical interface for the creation of RML rules.

Related studies

Some studies have been made to evaluate available tools and languages. [76] compared SPARQL-Generate implementation to RMLMapper. Their results showed that SPARQL-Generate has a better computational performance when transforming more than 1500 CSV rows in comparison with RMLMapper. They also concluded that SPARQL-Generate language is easier to learn and use for semantic web practitioners (who are likely already familiar with SPARQL), but this was based on a limited analysis of the cognitive complexity of query/mappings in the two languages. RMLEditor, a graphical tool to generate RML rules was proposed by [64]. They performed a usability evaluation for their tool with semantic web experts and non-experts. In the case of semantic web experts they also evaluate the differences between the textual approach (RML) and this new visual one. However, RMLEditor was neither compared with other similar tools nor RML with other languages. [65] proposed YARRRML as a human-readable text-based representation which offers an easier layer on top of RML and R2RML. However, the authors did not present any evaluation of this language. [84] made a comparative characteristic analysis of different mapping languages. However, a qualitative analysis is not performed and usability is only mentioned in NF1 ”Easy to use by Semantic Web experts” which only YARRRML and SPARQL-Generate achieve.

Thus, to the best of our knowledge no usability study was performed in these languages which share the easiness of use as one of their goals. Therefore, we introduce this study as a first step into the usability evaluation of heterogeneous data mapping languages.

2.3 Presentation of the languages under study

In this section we compare YARRRML, SPARQL-Generate and ShExML syntax by means of a simple example. These three tools each offer a DSL able to define mappings for heterogeneous data sources like we have seen in the previous section and their designers share the goal to be user friendly [84, 50]. RML and similar alternatives are not included in the comparison because they have a verbose syntax very close to the RDF data model. While it might be an interesting solution for users without any programming knowledge but familiar with RDF, we consider it more like a lower level middle language to compile to

[9] https://github.com/RMLio/yarrrml-parser

rather than a language to be used by programmers and data engineers. Indeed, YARRRML and ShExML engines are able to compile their mappings to RML.

For the sake of the example two small files on JSON and XML are presented in Listing 2.1 and Listing 2.2 respectively. Each one of these files define two films with 6 attributes—that could differ on name and structure—that will be translated to the RDF output showed in Listing 2.3. In this example, and with the aim to keep it simple, different ids are used in each entity; however, it is possible to use objects with same ids that could be merged into a single entity or divided into different new entities depending on users' intention.

Listing 2.1: JSON films file

```json
{
    "films": [
        {
            "id": 3,
            "title": "Inception",
            "date": "2010",
            "countryOfOrigin": "USA",
            "director": "Christopher Nolan",
            "screenwriter": "Christopher Nolan"
        },
        {
            "id": 4,
            "title": "The Prestige",
            "date": "2006",
            "countryOfOrigin": "USA",
            "director": "Christopher Nolan",
            "screenwriter": ["Christopher Nolan",
                "Jonathan Nolan"]
        }
    ]
}
```

Listing 2.2: XML films file

```xml
<films>
    <film id="1">
        <name>Dunkirk</name>
        <year>2017</year>
        <country>USA</country>
        <director>Christopher Nolan</director>
        <screenwriters>
            <screenwriter>Christopher Nolan</screenwriter>
        </screenwriters>
    </film>
    <film id="2">
        <name>Interstellar</name>
        <year>2014</year>
        <country>USA</country>
        <director>Christopher Nolan</director>
        <screenwriters>
            <screenwriter>Christopher Nolan</screenwriter>
            <screenwriter>Jonathan Nolan</screenwriter>
        </screenwriters>
    </film>
</films>
```

Listing 2.3: RDF output

```
@prefix :      <http://example.com/> .

:4      :country                "USA" ;
        :screenwriter           "Jonathan Nolan" ,
                                "Christopher Nolan" ;
        :director               "Christopher Nolan" ;
        :name                   "The Prestige" ;
        :year                   :2006 .

:3      :country                "USA" ;
        :screenwriter           "Christopher Nolan" ;
        :director               "Christopher Nolan" ;
        :name                   "Inception" ;
        :year                   :2010 .

:2      :country                "USA" ;
        :screenwriter           "Jonathan Nolan" ,
                                "Christopher Nolan" ;
        :director               "Christopher Nolan" ;
        :name                   "Interstellar" ;
        :year                   :2014 .

:1      :country                "USA" ;
        :screenwriter           "Christopher Nolan" ;
        :director               "Christopher Nolan" ;
        :name                   "Dunkirk" ;
        :year                   :2017 .
```

YARRRML

Listing 2.4: YARRRML transformation script for the films example

```
prefixes:
  ex: "http://example.com/"

mappings:
  films_json:
    sources:
      - [’films.json~jsonpath’, ’$.films[*]’]
    s: ex:$(id)
    po:
      - [ex:name, $(title)]
      - [ex:year, ex:$(date)~iri]
      - [ex:director, $(director)]
      - [ex:screenwriter, $(screenwriter)]
      - [ex:country, $(countryOfOrigin)]
  films_xml:
    sources:
      - [’films.xml~xpath’, ’//film’]
    s: ex:$(@id)
    po:
      - [ex:name, $(name)]
      - [ex:year, ex:$(year)~iri]
      - [ex:director, $(director)]
      - [ex:screenwriter, $(screenwriters/screenwriter)]
      - [ex:country, $(country)]
```

YARRRML is designed with human-readability in mind which is achieved through a YAML based syntax. Listing 2.4 shows the mappings `films_json` and `films_xml` for our films example. Each mapping starts with a source definition that contains the query to be used as iterator, e.g., `//film`. It is followed by the definition of the output given by a subject definition (`s:`) and a number of associated predicate-object definitions (`po:`). Subject and predicate-object definitions can use "partial" queries relative to the iterator to populate the subject and object values. This way of defining mappings is very close to RML; YARRRML actually does not provide an execution engine but is translated to RML.

SPARQL-Generate

Listing 2.5: SPARQL-Generate transformation script for the films example

```
BASE <http://example.com/>
PREFIX iter: <http://w3id.org/sparql-generate/iter/>
PREFIX fun: <http://w3id.org/sparql-generate/fn/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX : <http://example.com/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX schema: <http://schema.org/>
PREFIX sc: <http://purl.org/science/owl/sciencecommons/>

GENERATE {
  ?id_json :name ?name_json ;
       :year ?year_json ;
       :director ?director_json ;
       :country ?country_json .

  GENERATE {
    ?id_json :screenwriter ?screenwriter_json .
  }
  ITERATOR iter:Split(?screenwriters_json, ",")
        AS ?screenwriters_json_iterator
  WHERE {
    BIND(REPLACE(?screenwriters_json_iterator,
        "\\[|\\]|\"", "")
    AS ?screenwriter_json)
  } .

  ?id_xml :name ?name_xml ;
       :year ?year_xml ;
       :director ?director_xml ;
       :country ?country_xml .

  GENERATE {
    ?id_xml :screenwriter ?screenwriter_xml .
  }
  ITERATOR iter:XPath(?film_xml,
        "/film/screenwriters[*]/screenwriter")
        AS ?screenwriters_xml_iterator
  WHERE {
    BIND(fun:XPath(?screenwriters_xml_iterator,
    "/screenwriter/text()") AS ?screenwriter_xml)
  } .

}
ITERATOR iter:JSONPath(
        <https://raw.githubusercontent.com/herminiogg/ShExML/
        master/src/test/resources/filmsPaper.json>,
        "$.films[*]") AS ?film_json
ITERATOR iter:XPath(
        <https://raw.githubusercontent.com/herminiogg/ShExML/
        master/src/test/resources/filmsPaper.xml>,
        "//film") AS ?film_xml
WHERE {
  BIND(IRI(CONCAT("http://example.com/",
        STR(fun:JSONPath(?film_json,"$.id")))) AS ?id_json)
  BIND(fun:JSONPath(?film_json, "$.title") AS ?name_json)
  BIND(fun:JSONPath(?film_json, "$.director")
        AS ?director_json)
  BIND(IRI(CONCAT("http://example.com/",
        fun:JSONPath(?film_json, "$.date"))) AS ?year_json)
  BIND(fun:JSONPath(?film_json, "$.countryOfOrigin")
        AS ?country_json)
  BIND(fun:JSONPath(?film_json, "$.director")
        AS ?directors_json)
  BIND(fun:JSONPath(?film_json, "$.screenwriter")
    AS ?screenwriters_json)
  BIND(IRI(CONCAT("http://example.com/",
        fun:XPath(?film_xml,"/film/@id"))) AS ?id_xml)
  BIND(fun:XPath(?film_xml, "/film/name/text()")
        AS ?name_xml)
  BIND(fun:XPath(?film_xml, "/film/director/text()")
        AS ?director_xml)
  BIND(IRI(CONCAT("http://example.com/",
        fun:XPath(?film_xml, "/film/year/text()")))
                AS ?year_xml)
  BIND(fun:XPath(?film_xml, "/film/country/text()")
        AS ?country_xml)
}
```

SPARQL-Generate is an extension of SPARQL 1.1 for querying heterogeneous data sources and creating RDF and text. It offers a set of SPARQL binding functions and SPARQL iterator functions to achieve this goal. The mapping for our films example is shown in Listing 2.5. The output of the mapping is given within the GENERATE clauses and can use variables and IRIs, while queries, IRI and variable declarations are declared in the WHERE clause. SPARQL-Generate is an expressive language that can be further extended using the SPARQL 1.1 extension system. On the other side, SPARQL-Generate scripts tend to be verbose compared to the other two languages studied in this paper.

ShExML

Listing 2.6: ShExML transformation script for the films example

```
PREFIX : <http://example.com/>
SOURCE films_xml_file <
        https://raw.githubusercontent.com/herminiogg/
        ShExML/master/src/test/resources/filmsPaper.xml>
SOURCE films_json_file <
        https://raw.githubusercontent.com/herminiogg/
        ShExML/master/src/test/resources/filmsPaper.json>
ITERATOR film_xml <xpath: //film> {
    FIELD id <@id>
    FIELD name <name>
    FIELD year <year>
    FIELD country <country>
    FIELD director <director>
    FIELD screenwriters <screenwriters/screenwriter>
}
ITERATOR film_json <jsonpath: $.films[*]> {
    FIELD id <id>
    FIELD name <title>
    FIELD year <date>
    FIELD country <countryOfOrigin>
    FIELD director <director>
    FIELD screenwriters <screenwriter>
}
EXPRESSION films <films_xml_file.film_xml
        UNION films_json_file.film_json>

:Films :[films.id] {
    :name [films.name] ;
    :year :[films.year] ;
    :country [films.country] ;
    :director [films.director] ;
    :screenwriter [films.screenwriters] ;
}
```

ShExML, our proposed language, can be used to map XML and JSON documents to RDF. The ShExML mapping for the films example is presented in Listing 2.6. It consists of source definitions followed by iterator definitions. The latter define structured objects which fields are populated with the results of source queries. The output of the mapping is described using a Shape Expression (ShEx) [105, 17] which can refer to the previously defined fields. The originality of ShExML, compared to the other two languages studied here, is that the output is defined only once even when several sources are used. This is a design choice that allows the user to separate concerns: how to structure the output on the one hand, and how to extract the data on the other hand.

Comparing languages features

In this subsection we compare languages features and what operations are supported or not in each language (see Table 2.1).

Iterators, sources, fields, unions and so on are common to the three languages as they have the same objective. They have different syntaxes, as it can be seen in the three examples, but from a functionality point of view there are no differences.

Source and output definition and their artefacts: As we saw, the mechanism to define the form of the RDF output has different flavour in the three languages: subject and predicate-object definitions for every source in YARRRML; GENERATE clauses for every source in SPARQL-Generate; a single Shape Expression in ShExML. Additionally, the three languages offer slightly different operators for constructing the output values. All of them typically obtain IRIs by concatenating a source value to some prefix, and reuse literal values as is. YARRRML supports the generation of multiple named graphs whereas SPARQL-Generate can only generate one named graph at a time and ShExML only generates RDF datasets.

Multiple results: The handling of multiple results, like it occurs on the screenwriters case, is different between SPARQL-Generate and the two other languages. In YARRRML and ShExML if a query returns multiple results they are treated like a list of them. However, in SPARQL-Generate this functionality must be explicitly declared like it can be seen in Listing 2.5. It leads to complex iterator definitions like the one used in JSON screenwriters one.

Transformations: The possibility of transforming the output to another value by means of a function is something very useful for different purposes when building a knowledge graph. Therefore, in YARRRML this is supported through the FnO mechanism [85] which offers a way to define functions inside mapping languages in a declarative fashion. SPARQL-Generate offers some functions for strings embedded inside the SPARQL binding functions mechanism; however, it is possible to extend the language through the SPARQL 1.1 extension mechanism. In the case of ShExML, only Matchers and String operations are offered for transformation purposes.

Other formats output: Output format on YARRRML and ShExML is limited to RDF; whereas, in SPARQL-Generate it is possible to also generate plain text, enabling the potential transformation to a lot of different formats. In this aspect, SPARQL-Generate presents a much more flexible output. Conversely, YARRRML and ShExML engines offer a translation of their mappings to RML rules which improves interoperability with other solutions.

Link to other mappings: In YARRRML there is the possibility to link mappings between them. This functionality is provided by giving the name of the mapping to be linked and the condition that must be satisfied (e.g., ID of mapping A equal to ID of mapping B). This can be useful when the subject is generated with a certain attribute but this attribute does not appear on the other file so the linking should be done using another attribute. In ShExML this can be partially achieved by Shape linking—which is a syntactic sugar to avoid repeating an expression twice—and by the Join clause which gives an implementation for primary interlinking covering a subset of what is covered with YARRRML mapping linking. In SPARQL-Generate this can be achieved using nested Generate clauses and Filter clauses.

Conditional mapping generation: Sometimes there is the need to generate triples only in the case that some condition is fulfilled. In YARRRML this is achieved using the conditional clause and a function. In SPARQL-Generate this can be obtained with the SPARQL 1.1 Filter clauses and also with the extensibility mechanism offered by the language. In ShExML this is not possible currently.

Further features of SPARQL-Generate: Apart from what has been presented in the previous point, SPARQL-Generate, as being based on SPARQL 1.1, offers more expressiveness than the other two languages. One possibility that emerges from that is the use of defined variables. For example, it is possible to define an iterator of numbers and then use that numbers to request different parts of an API. This versatility enables the creation of very complex and rich scripts that can cover a lot of use cases. It is natural to expect that learning to use the full capabilities of SPARQL-Generate is complex, as the language offers a lot of features. In our experiments, however, only some basic features of the language were required and, as is shown in Section 2.5, it appears that SPARQL-Generate design did not help test subjects to solve the proposed tasks easily.

Features		ShExML	YARRRML	SPARQL-Generate
Source and output definition	**Defining output**	Shape expression	Subject and predicate-object definitions	Generate clause
	IRIs generation	Prefix and value generation expression (concatenation)	Prefix and value generation expression (array)	Variable (previous use of concat function) or string interpolation
	Datatypes & Language tags	Yes	Yes	Yes
Multiple results from a query		Treated like an array	Treated like an array	Need to iterate over the results
Transformations		Limited (Matchers and String operators).	FnO hub	Functions for strings and extension mechanism
Output formats	**Output**	RDF	RDF	RDF and any text-based format
	Translation	RML	RML	No translation offered
Link between mappings		Shape Linking and JOIN keyword (do not fully cover YARRRML feature)	Yes (conditions allowed)	Nested generate clauses, filter clauses and extension mechanism
Conditional mapping generation		No	Yes (Function and conditional clause)	Yes (Filter clause and extension mechanism)

Table 2.1: Features comparison between the three languages

2.4 Methodology

In order to test our hypothesis that ShExML is easier for first-time users only experienced in programming and the basics of linked data, an experiment was carried out. The University of Oviedo granted ethical approval to carry out the described study. Verbal consent was requested before starting the experiment.

Experiment design

The selected tools were YARRRML[10], SPARQL-Generate[11] and ShExML[12]. We decided not to include RML[13] and similar alternatives for the same reason mentioned on Section 2.3. Three manuals were designed for the students based on the example about films that described how the integration can be done with each tool[14]. The experiment was designed to be performed in each tool dedicated online environment, which are available through the Internet as a webpage.

In addition, a small manual was developed to guide the students along the experiment and to inform them about the input files and which are the expected outputs[14]. This manual contained two tasks to perform during the experiment which were designed to be performed sequentially, i.e., the student should finish the first task before starting with the second one. The first task was the mapping and integration of two files (JSON and XML) with information about books which should be mapped in a unique RDF graph. The final output should be equal to the one given in the guide. The second task was to modify the script done in the previous task so that the prices are separated and can be compared between markets. In other words, that multiple prices are tagged individually referring to the market where the specific price was found, like they were in the input files. This second task gives us an intuition on how easy is to modify an existing set of data mapping rules in each language.

The study was designed as a mixed method approach, including a quantitative analysis and a qualitative analysis. For the quantitative analysis measures, Mousotron[15] was used which allows to register the number of keystrokes, the

[10] http://rml.io/yarrrml/

[11] https://ci.mines-stetienne.fr/sparql-generate/

[12] http://shexml.herminiogarcia.com/

[13] http://rml.io/

[14] Material can be consulted on: https://github.com/herminiogg/shexml-paper-2019-data/tree/master/experiment-material

[15] http://www.blacksunsoftware.com/mousotron.html

distance travelled by the mouse and so on. For the qualitative analysis two Office 365 forms were used with questions based on a Likert scale (see questions in Table 2.2). In addition, the elapsed time was calculated from timestamps in the Office 365 forms.

Conduction

The sample consisted on 20 students (4 women and 13 men) of the MSc in Web Engineering first-year course (out of two years) at the University of Oviedo[16]. Most of them have a bachelor degree (240 ECTS credits) in computer science or similar fields. They were receiving a semantic web course of two weeks—a total of 30 hours (3 hours per day)— where they were introduced to semantic technologies like: RDF, SPARQL, ShEx, etc. Before this course they had not previous knowledge on semantic web technologies. Regarding prior knowledge of YAML by subjects, even though it is normally known and used by developers, we could not assure it. The experiment was hosted the final day of the course.

The experiment was conducted in their usual classroom and with their whole-year-assigned computers. So that they were in a confortable environment and with a computer they are familiar with. The three tools were assigned to the students in a random manner. Each student received the printed manual for its assigned tool and they were given a time of 20 minutes to read it, test the language in the online environment, and ask doubts and questions. Once these 20 minutes were elapsed the printed experiment guide was given to the students and they were explained about the experiment proceeding with indications about Mousotron operation.

In particular the procedure followed to perform the whole experiment was:

1. Open the assigned tool on the dedicated webpage and clear the given example.
2. Open Mousotron and reset it.
3. Proceed with task 1 (start time registered for elapsed time calculation).
4. Once task 1 is finished, capture Mousotron results (screenshot) and fill the first Office 365 questionnaire.
5. Reset Mousotron and proceed with task 2.
6. Once task 2 is finished, capture Mousotron results (screenshot) and fill the second Office 365 questionnaire.

Analysis

The quantitative results were dump into an Excel sheet and anonymised. Although many results can be used as given by the students, some of them need to be calculated. This is the case of elapsed time (on both tasks), completeness percentage and precision. Elapsed time in the first task (t_{t1}) was calculated as the subtraction of questionnaire 1 beginning time (st_{q1}) and experiment start time (st_e), i.e., ($t_{t1} = st_{q1} - st_e$). Elapsed time in the second task (t_{t2})

[16] `http://miw.uniovi.es/`

was calculated as the subtraction of questionnaire 1 ending time (et_{q1}) and questionnaire 2 beginning time (st_{q2}), i.e., ($t_{t2} = st_{q2} - et_{q1}$).

Completeness percentage was calculated from three measures: the proportion of correctly generated triples contributed 50%, the proportion of data correctly translated contributed 25% and the proportion of correctly generated prefixes and datatypes as a 25%. This design gives more importance to the structure, which is the main goal when using these tools. Other aspects, like correct data (i.e., the object part of a triple), prefixes (i.e., using the correct predicate for the subject, the predicate and the object in case of an IRI) and the datatype (i.e., putting the correct xsd type in case of a literal object) are a little less valued as these errors could come more easily from a distraction or an oversight. Let CP be the completeness percentage, t the number of triples, d the number of data gaps and $p\&dt$ the number of prefixes and datatypes, so the calculation of the completeness percentage can be expressed as:

$$CP = 0.5 * \frac{t_{total} - t_{generated}}{t_{total}} + 0.25 * \frac{d_{total} - d_{generated}}{d_{total}} + 0.25 * \frac{p\&dt_{total} - p\&dt_{generated}}{p\&dt_{total}}$$

Finally, precision was calculated as the division of current student elapsed time by minimum elapsed time of all students, multiplied by the completeness percentage. This precision formulation gives us an intuition on how fast was some student in comparison with the fastest student and with a correction depending on how well his/her solution was. Let t_{sn} be the elapsed time of student n and CP_{sn} the completeness percentage of student n calculated with the previous formula.

$$Precision_{sn} = \frac{t_{sn}}{min(\{t_{s1}, ..., t_{sn}\})} * CP_{sn}$$

The results of the qualitative analysis were only anonymised as they can be directly used from the Office 365 output.

For the analysis the IBM SPSS version 24 was used. We planned a One Way ANOVA test within the three groups in the quantitative analysis where a normal distribution was found and the Kruskal-Wallis test where not. The qualitative analysis comparison between three groups was established using the Kruskal-Wallis test. The report and analysis of the results was made using [44] as guidance and using the suggested APA style as a standard manner to report statistical results.

Threat to validity

In this experiment we have identified the following threats to its validity.

Internal validity

We have identified the following internal validity threats in the experiment design:

- More expertise in some specific tool: In semantic web area—as in other areas—people tend to be more expert in some specific technologies and languages. The derived risk is that this expertise can have an influence on final results. To alleviate this we have selected MSc students that

Table 2.2: Statements to evaluate by the students based on a 5 point Likert scale

Questionnaire	Statement	Obtained Variable
1	The experience with the tool was satisfactory	General satisfaction level
1	The tool was easy to use	Easiness of use
1	The mapping definitions was easy	Mapping definition easiness
1	The language was easy to learn	Learnability
1	I find that these tool can be useful in my work	Applicability
1	The coding in this tool was intuitive	Intuitiveness
1	The language design leads to commit some errors	Error proneness
1	The error messages were useful to solve the problems	Error reporting usefulness
2	It was easy to define different predicates for the price	Modifiability

are studying the same introductory semantic web course and we have assigned the tools in a random manner.

- Not homogeneous group: It is possible that the selected group is not homogeneous on skills and previous knowledge. To mitigate this we have applied the same measures as for the previous threat: Students of a semantic web course and a randomised tool assignment.

- Unfamiliar environment: In usability studies, unfamiliar environments can play a role on final conclusions. Therefore, we opted to run the experiment in a well-known environment for the students, that is, their whole-year classroom.

- More guide and information about one tool: As we have designed one of the languages, it could lead to a bias in information delivery. To try to mitigate this threat we developed three identical manuals for each tool. Questions and doubts were answered equally for all the students and tools.

External validity

Following the measures taken in the internal validity threats we identified the corresponding external validity ones:

- Very focused sample: As we have restricted the profile of the sample to students of a MSc course which are more or less within the same knowledge level, there is the risk that these findings cannot be extrapolated for other samples or populations. It is possible that for semantic web practitioners—with different interests and expertises—these findings are not applicable. However, the intention of this study was to evaluate usability with first-time users as a first step to guide future studies.

2.5 Results

From the 20 students of the sample[17], in the first task, 3 of them left the experiment without making any questionnaire, 2 for SPARQL-Generate and 1

[17] Original datasets available on:
https://github.com/herminiogg/shexml-paper-2019-data/tree/master/datasets

for YARRRML. In the second task, only 7 out of the 20 students made the questionnaire, 6 for ShExML and 1 for YARRRML. The statistical analysis was made using the IBM SPSS software, version 24.

Task 1: As previously stated, the number of students that finished—correctly or not—the proposed task was 17. Descriptive statistics can be seen in Table 2.3. Comparison of three groups was made by means of a One Way ANOVA which results showed significant differences on elapsed seconds $F(2, 14) = 6.00, p = .013, \omega = .60$. As completeness percentage and precision are not following a normal distribution on SPARQL-Generate group ($W(4) = .63, p = .001$ and $W(4) = .63, p = .001$), the comparison was established by means of the Kruskal-Wallis test which showed significant differences in both variables ($H(2) = 9.73, p = .008$ and $H(2) = 9.68, p = .008$). Post hoc test for elapsed seconds using the Gabriel's criterion showed significant differences between ShExML group and YARRRML group ($p = .016$). Post hoc test for completeness percentage and precision using the Bonferroni's criterion showed significant differences between ShExML and SPARQL-Generate ($p = .012, r = .87$ and $p = .012, r = .87$). Likert scale questionnaire results ($\alpha = 0, 73$) (see Fig. 2.1) were analysed using Kruskal-Wallis test which resulted in significant differences between groups for variables general satisfaction level ($H(2) = 6.28, p = .043$), easiness of use ($H(2) = 9.82, p = .007$), mapping definition easiness ($H(2) = 10.25, p = .006$) and learnability ($H(2) = 8.63, p = .013$). Bonferroni's criterion was used as post hoc test for the variables with significant differences. For general satisfaction level significant differences were found between ShExML and YARRRML ($p = .039, r = .69$). For easiness of use significant differences were found between ShExML and YARRRML ($p = .011, r = .81$). For mapping definition easiness significant differences were found between ShExML and SPARQL-Generate ($p = .013, r = .90$) and between ShExML and YARRRML ($p = .037, r = .69$). For learnability significant differences were found between ShExML and SPARQL-Generate ($p = .042, r = .78$) and between ShExML and YARRRML ($p = .040, r = .69$).

Task 2: In this task only 7 students reached this step: 6 for ShExML and 1 for YARRRML. Descriptive statistics of this task can be seen in Table 2.4. No significant differences were found in any of the variables. In subjective variable analysis (see Fig. 2.2) no significant differences were found.

2.6 Discussion

Statistical results discussion

Results of task 1 show that variables like keystrokes, left button clicks, right button clicks, mouse wheel scroll and meters travelled by the mouse, do not have a significant variability depending on the used tool. This suggests that web interfaces used as online development environments are more or less homogeneous and do not have an impact on the development of the scripts. However, keystrokes variable results should be considered with caution because for SPARQL-Generate the mean of completeness percentages was very low; therefore, achieving a final solution may involve more keystrokes. On the other hand, elapsed seconds, completeness percentage and precision show significant differences between groups which suggest that the selected language has an influence on these variables. Moreover, we can see that elapsed seconds has a

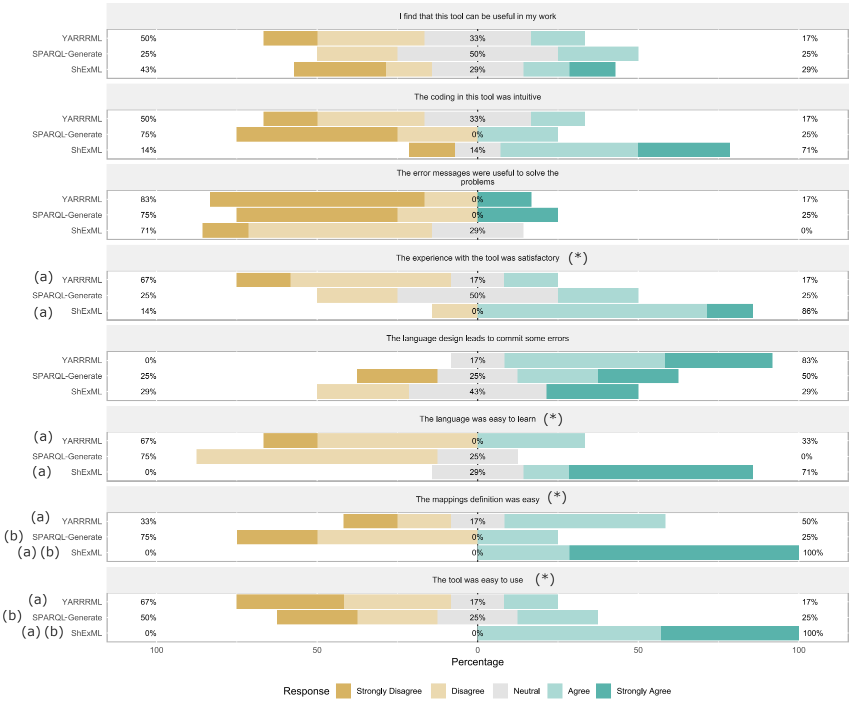


Figure 2.1: Task 1 results for Likert scale questionnaire where results are divided into questions and groups. (*) means significant differences between groups and (a) and (b) means significant differences in the post hoc test between the marked groups at the level of significance ($\alpha = .05$)

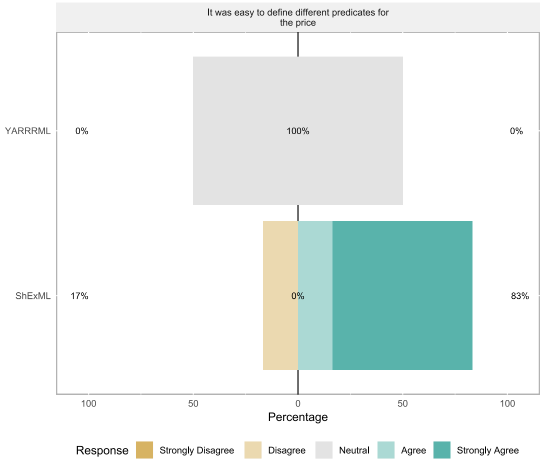


Figure 2.2: Task 2 results for Likert scale questionnaire where results are divided into the two groups.

Table 2.3: Descriptive statistics for task 1 objective results where n is the sample size, $\bar{x}$ is the mean, s is the standard deviation, max is the maximum value of the sample and min is the minimum value of the sample. (*) means significant differences between groups and (a) means significant differences in the post hoc test between the marked groups at the level of significance ($\alpha = .05$). Differences in totals are due to malfunctions while operating capture software.

Measure	Group	n	$\bar{x}$	s	max	min
Elapsed seconds (*)	ShExML (a)	7	1560.1429	541.57376	2192	782
	YARRRML (a)	6	2443.8333	375.44502	2896	1891
	SPARQL-Generate	4	2292.7500	533.49063	2769	1634
	Total	17	2044.4118	620.68370	2896	782
Keystrokes	ShExML	6	1138.50	610.588	2287	674
	YARRRML	4	1187	449.649	1795	810
	SPARQL-Generate	3	1125.67	121.476	1265	1042
	Total	13	1150.46	457.183	2287	674
Left button clicks	ShExML	6	176.50	112.169	327	58
	YARRRML	4	318.75	177.989	551	170
	SPARQL-Generate	3	166	78.791	254	102
	Total	13	217.85	138.267	551	58
Right button clicks	ShExML	6	2.17	2.137	6	0
	YARRRML	4	2.25	1.708	4	0
	SPARQL-Generate	2	4.50	2.121	6	3
	Total	12	2.58	2.021	6	0
Mouse wheel scroll	ShExML	6	148	183.737	486	13
	YARRRML	4	679.25	606.711	1404	101
	SPARQL-Generate	3	199	131.160	348	101
	Total	13	323.23	412.819	1404	13
Meters travelled by the mouse	ShExML	7	30.400	24.318	70.079	0
	YARRRML	6	43.454	43.144	101.767	0
	SPARQL-Generate	4	21.220	16.526	37.680	0
	Total	17	32.847	30.550	101.767	0
Completeness percentage (*)	ShExML (a)	7	0.771	0.296	1	0.19
	YARRRML	6	0.323	0.366	0.82	0
	SPARQL-Generate (a)	4	0.02	0.04	0.08	0
	Total	17	0.436	0.415	1	0
Precision (*)	ShExML (a)	7	0.495	0.286	1	0.07
	YARRRML	6	0.131	0.160	0.38	0
	SPARQL-Generate (a)	4	0.005	0.01	0.02	0
	Total	17	0.251	0.292	1	0

medium size effect ($\omega = .60$). Post hoc results show that there are significant differences between ShExML and YARRRML which suggests that YARRRML users tend to need more time than ShExML users for these tests. In the case of comparisons with SPARQL-Generate there are not significant differences which can be due to the small sample size and the low completeness percentage. Differences between ShExML and SPARQL-Generate for completeness percentage and precision suggest that SPARQL-Generate users were not able to achieve working solutions as ShExML users, which have the highest mean on both variables. However, between ShExML and YARRRML groups there were no significant differences which is in line with the great variability of those two variables.

Results of task 2 do not show any significant difference between the ShExML group and the YARRRML group. This can be explained by the low sample

Table 2.4: Descriptive statistics for task 2 objective results where n is the sample size, $\bar{x}$ is the mean, s is the standard deviation, max is the maximum value of the sample and min is the minimum value of the sample. Differences in totals are due to malfunctions while operating capture software.

Measure	Group	n	$\bar{x}$	s	max	min
Elapsed seconds	ShExML	6	325.5	328.9248	879	3
	YARRRML	1	47	0	47	47
	Total	7	285.7143	318.1822	879	3
Keystrokes	ShExML	5	206.40	175.832	438	43
	YARRRML	1	91	0	91	91
	Total	6	187.17	164.174	438	43
Left button clicks	ShExML	5	61.80	81.417	207	16
	YARRRML	1	43	0	43	43
	Total	6	58.67	73.225	207	16
Right button clicks	ShExML	5	0.40	0.548	1	0
	YARRRML	1	0	0	0	0
	Total	6	0.33	0.516	1	0
Mouse wheel scroll	ShExML	5	123.80	129.494	288	0
	YARRRML	1	41	0	41	41
	Total	6	110	120.655	288	0
Meters travelled by the mouse	ShExML	6	9.7629	13.8829	37.7565	0
	YARRRML	1	11.7563	0	11.7563	11.7563
	Total	7	10.0477	12.6957	37.7565	0
Completeness percentage	ShExML	6	0.73	0.3904	1	0
	YARRRML	1	0	0	0	0
	Total	7	0.6257	0.4507	1	0
Precision	ShExML	6	0.4683	0.37467	1	0
	YARRRML	1	0	0	0	0
	Total	7	0.4014	0.38512	1	0

size in the YARRRML group where only one individual made this step. However, completeness percentage and precision show us that some students did achieve a correct solution with ShExML, whereas in YARRRML group and in SPARQL-Generate group they did not. This leads to the conclusion that only the ShExML group managed to find a working solution to both proposed tasks. Nevertheless, these conclusions must be validated with bigger experiments to have statistical confidence.

The differences in completeness percentage and precision between ShExML and SPARQL-Generate and also between ShExML and YARRRML in elapsed seconds can lead us to the conclusion that usability on first-time users is improved by using ShExML over the other two languages, which answers RQ1. Moreover, this conclusion is reinforced by the situation that in task 2 neither YARRRML nor SPARQL-Generate users were able to find a solution to this task.

Regarding the subjective analysis, significant differences were found between groups in general satisfaction level, mapping definition easiness, easiness of use and learnability (as perceived by the students).

On general satisfaction level significant differences were found between ShExML and YARRRML which indicates that ShExML users were more satisfied with the overall use of the tool respect to the YARRRML users. Differences between SPARQL-Generate users and the two other groups could not be established due to their low completeness percentage and precision rates.

In the case of easiness of use significant differences were found between

ShExML and YARRRML which suggests that ShExML users found this language easier to use than YARRRML users did with their language counterpart. In this case, like in the previous variable, significant differences could not be established between SPARQL-Generate and the two other groups due to low completeness percentage. In mapping definition easiness differences were established between ShExML group and YARRRML group and between ShExML group and SPARQL-Generate group which indicates that ShExML users found mappings easier to define in ShExML than in the other two languages. We also note that users did not find differences on mapping definition easiness between YARRRML and SPARQL-Generate, this may be because SPARQL-Generate users did not use the whole language.

On learnability significant differences were found between ShExML and SPARQL-Generate and between ShExML and YARRRML which suggests that the users found easier to learn ShExML than the other two languages. However, no significant differences were found between YARRRML and SPARQL-Generate which seems strange due to the difference of verbosity between the two languages.

Differences on subjective analysis between ShExML and YARRRML on general satisfaction level, mapping definition easiness, easiness of use and learnability, and between ShExML and SPARQL-Generate on mapping definition easiness and learnability comes to corroborate what we have elucidated with the objective analysis answering RQ1.

Review of the other variables shows that the users do not see much applicability on the three languages, that the design of the languages leads users to commit some errors during the development of the script and that the error reporting system in the three of them is not very useful to solve the incoming problems.

The feedback received from the users in the error proneness and error reporting usefulness variables determines that these two aspects are the ones that should be improved in the three languages to improve their usability. This comes to answer the RQ3.

For the modifiability variable assessed in task 2, ShExML users tend to rate this feature with high marks whereas the single YARRRML user gave a response of 3 in a 5 point Likert scale which is in line with his/her completeness percentage mark. As with the objective results of task 2, these subjective results should be further validated in future bigger experiments to corroborate these early findings.

Alignment with features comparison

In the light of the statistical analysis outcome, SPARQL-Generate design has been shown to have a negative impact on first-time users. This led to three users abandoning the task and low completeness scores for the rest of the group. Although having more features in a language is something good and desirable, these results caught attention on how these features should be carefully designed and included in the language in order to improve easiness of use, and thus overall adoption of the tool. In the case of YARRRML language, although it has been designed with human-friendliness in mind, in our experiment it has not reached the expected results in comparison with ShExML. However, it has better results than SPARQL-Generate, suggesting it is less complex to use than that language, but still more complex than ShExML. Nevertheless, it does not seem that supported features could explain the differences between YARRRML and ShExML as the features used on the experiment are more or less equal. Instead other syntax details may be affecting the differences between these two groups such as: the use of keywords that made the language more self explanatory and the modularity used on iterators which reminds of object-oriented programming languages. However, this would require a broader study taking into account programming style background of participants and their own style preferences using techniques like a cognitive complexity architecture [62] to identify how each feature and its design is affecting the usability of each specific language.

These results highlight the importance on how features are designed and included in a language. Therefore, SPARQL-Generate with more features and being a highly flexible language tends to have a bad influence on users' usability. Comparing ShExML and YARRRML we see that these differences are smaller than with SPARQL-Generate and that features support does not seem to be the variable affecting YARRRML usability. Thus, we can conclude—and answer the RQ2—that it is not the features supported by a language which affects usability of first-time users but their design.

2.7 Conclusions and Future Work

In this work we have compared the usability of three heterogeneous data mapping languages. The findings of our user study were that better results, and speed on finding this solution, are related to ShExML users whereas SPARQL-Generate users were not able to find any solution under study conditions. In the case of YARRRML users, they performed better than SPARQL-Generate users but worse than ShExML users finding partial solutions to the given problem.

This study is (to our knowledge) the first to explore the topic of usability for first-time users with programming and Linked Data background in these kind of languages. It also reflects the importance that usability has on the accuracy of the encountered solutions and how features should be carefully designed in a language to not impact negatively on its usability.

As future work, bigger experiments should be carried out with an emphasis on programming style background and styles (using cognitive complexity frameworks) to corroborate and expand these early findings. In addition, improving these aspects that were worst rated in the three languages (i.e., error proneness and the error reporting system) would enhance perceived user friendliness.

This work highlights the importance of usability on these kind of languages and how it could affect their adoption.

Acknowledgements

We want to thank the students of the Master's Degree in Web Engineering for their willingness to participate in the experiment described in this work.

Funding

This work has been funded by the Principality of Asturias through the Severo Ochoa call (grant BP17-29), by the Ministry of Economy, Industry and Competitiveness under the call of ”Programa Estatal de I+D+i Orientada a los Retos de la Sociedad” (project TIN2017-88877-R), the CPER Nord-Pas de Calais/FEDER DATA Advanced data science and technologies 2015-2020, and the ANR project DataCert ANR-15-CE39-0009. There was no additional external funding received for this study.

Chapter 3

XMLSchema2ShEx: Converting XML validation to RDF validation

This article was originally published as:
Herminio García-González and José Emilio Labra Gayo. XMLSchema2ShEx: Converting XML validation to RDF validation. *Semantic Web*, 11(2):235–253, 2020

Herminio García-González and José Emilio Labra Gayo were with the Department of Computer Science, University of Oviedo, Oviedo, Asturias, Spain

The journal has the following metrics according to 2019 JCR:

- 2019 Impact Factor: 3.182
- 5 Year Impact Factor: 3.055
- Computer Science, Artificial Intelligence: Q2 (46/137)
- Computer Science, Information Systems: Q2 (49/156)
- Computer Science, Theory & Methods: Q1 (22/108)

Abstract

RDF validation is a field where the Semantic Web community is currently focusing attention. Besides, there is a recent trend to migrate data from different sources to semantic web formats. Therefore, in order to facilitate this transformation, we propose: a set of mappings that can be used to convert from XML Schema to Shape Expressions (ShEx), a prototype that implements a subset of the proposed mappings, an example application to obtain a ShEx schema from an XML Schema and a discussion on conversion implications of non-deterministic schemata. We demonstrate that an XML and its corresponding XML Schema are still valid when converted to their RDF and ShEx counterparts. This conversion, along with the development of other format mappings, could drive to an improvement of data interoperability due to the reduction of the technological gap.

3.1 Introduction

Data validation is a key area when normalisation and confidence are desired. Normalisation—which can be defined, in this context, as using an homogeneous schema or structure across different sources of similar information—is desired as a way of making a dataset more reliable and even more useful to possible consumers because of its standardised schema. Validation can excel data cleansing, querying and standardisation of datasets. In words of P.N. Fox et al. [47]: *"Procedures for data validation increase the value of data and the users' confidence in predictions made from them. Well-designed data management systems may strengthen data validation itself, by providing better estimates of expected values than were available previously."*. Therefore, validation is a key field of data management.

XML Schema [12] was designed as a language to make XML validation possible with more expressiveness than DTDs [11]. Using XML Schema developers can define the structure, constraints and documentation of an XML vocabulary. Besides DTD and XML Schema, other alternatives for XML validation (such as Relax NG [24] and Schematron [66]) were proposed.

In the Semantic Web, RDF was missing a standard constraints validation language which covers the same features that XML Schema does for XML. Some alternatives were OWL [56] and RDF Schema [20]; however, they do not cover completely what XML Schema does for XML [123]. For this purpose, Shape Expressions (ShEx) [104, 106] was proposed to fulfill the requirement of a constraints validation language for RDF, and SHACL [69] (another proposed language for RDF validation) has recently become a W3C recommendation.

As many documents and data are persisted in XML, the need for migration and interoperability to more flexible data is nowadays more pressing than ever, many authors have proposed conversions from XML to RDF [92, 35, 7, 14], with the goal of transforming XML data to Semantic Web formats.

Although these conversions enable users to migrate their data to Semantic Web, means for validating the output data after converting XML to RDF are missing. Therefore, we should ensure that the conversion has been done correctly and that both versions—in different languages—are defining the same meaning.

Conversions between XML and RDF, and between XML Schema and ShEx are necessary to alleviate the gap between semantic technologies and more traditional ones (e.g., XML, JSON, CSV, relational databases). With that in mind, providing generic transformation tools from non-semantic technologies to semantic technologies can enhance the migration possibilities; in other words, if we can create tools that ease the transformation and adaptation among technologies we will encourage future migrations. Taking Text Encoding Initiative (TEI) [39] as an example, digital humanities can take benefit from Semantic Web approaches [121, 115]. There are many manuscripts transcribed to XML—using TEI—that can be converted to RDF. But transcribers are hesitant to deal with the underlying technology although they can benefit from it [87]. Those are the cases where generic approaches, as the one introduced here, can offer a solution and where automatic conversion of schemata has its place when transformations are to be checked.

Taking into account what we previously presented, the questions that we want to address in the present work are the following:

- RQ1: What components should have a mapping from XML Schema to ShEx?
- RQ2: How to ensure that both schemata are equivalent?
- RQ3: Is it possible to ensure a backwards conversion in all cases?
- RQ4: Are non-deterministic schemata (i.e., ambiguous schemata) possible to translate and validate?

In this paper, we describe a solution on how to make the conversion from XML Schema to ShEx. We describe how each element in XML Schema can be translated into ShEx. Moreover, we present a prototype that can convert a subset of what is defined in the following sections.

The rest of the paper is structured as follows: Section 3.2 presents the background; Section 3.3 gives a brief introduction to ShEx; Section 3.4 describes a possible set of mappings between XML Schema and ShEx; Section 3.5 presents a prototype used to validate a subset of previously presented mappings and how this conversion works against existing RDF validators; Section 3.6 discusses the implications of Non-Deterministic schemata on our work. Finally, Section 3.7 draws some conclusions and future lines of work and improvement.

3.2 Background

The related work of XML ecosystem conversion can be divided in three main categories: conversions from XML to Semantic Web formats, conversions from XML schemata to non Semantic Web schemata and conversions from XML schemata to RDF schemata.

From XML to Semantic Web formats

Along with schemata conversions, data transformation has to be tackled. Therefore many authors have worked on this topic of converting from XML to Semantic Web formats and more specifically to RDF. For this conversions there are plenty of strategies that have been proposed and followed by other authors.

In [92], authors describe their experience on developing this transformation for business to business industry in the case of the Semantic Mediation tools. An XML Schema to RDF Schema transformation is performed as part of the requirement of the Semantic Mediation tool.

In [35], a transformation between XML and RDF depending on an ontology is described. This transformation takes an XML document, a mapping document and an ontology document and makes the transformations to RDF instances compliant with the input ontology. Using the mapping file, conversions between the XML Schema and the ontology are established.

In [5], the author explains how XML can be converted to RDF—and vice versa—using XML Schema as the base for the mappings. This work is then expanded in [7] where the author tries to solve the lift problem (the problem of how to map heterogeneous data sources in the same representational framework) from XML to RDF and backwards by using the Gloze mapping approach on top of Apache Jena.

In [128], the authors present a mechanism to query XML data as RDF. Firstly, a matching from XML Schema to RDF Schema class hierarchy is performed. Then XML elements can be interpreted as RDF triples. The same procedure but using DTDs is described in [126].

In [19], the author presents a technique for making standard transformations between XML and RDF using XSLT. A case study in the field of astronomy is used to illustrate the solution.

Another approach using XSLT is [118] where authors describe a mapping mechanism using XSLT that can be attached to schemata definition.

In [9], a transformation from RDF to other kind of formats, including XML, is proposed using in XSLT stylesheets embedded SPARQL which by means of these extensions, could query, merge and transform data from the Semantic Web.

In [14], authors describe XSPARQL which is a framework that enables the transformation between XML and RDF based on XQuery and SPARQL and solves the disadvantages of using XSLT for these transformations.

However, these works (except [92]) are not covering the schemata mapping problem.

From XML schemata to other schemata

Although data migration is important, during this process it is desirable to transform the constraint rules or schemas too. This is also a way to verify that the transformations have been done correctly. Therefore, many authors have proposed different techniques and transformation from XML Schema.

In [97], a transformation from XML Schema to JSON Schema is proposed. These transformations are made using equivalent constraints when it is possible and concrete transformations when no equivalent constraints exists.

In [101], an algorithm that converts from XML Schemata to ER diagrams is proposed. This algorithm (called Xere mapping) is proposed as a part of the Xere technique to assist the integration of XML data.

In [74], the authors propose an algorithm to convert from a relational schema to an XML Schema and two algorithms to convert from a XML Schema to a relational schema. All these techniques preserve the structure and the semantics.

However, none of these works bring XML schemata to Semantic Web technologies.

From XML schemata to RDF schemata

In the Semantic Web community there has been an effort to convert XML schemata to OWL [43, 109] and to RDF Schema [92]. Moreover, when no schema is available the transformation can be performed from XML to OWL [16, 70, 98, 73].

However, RDF Schema and OWL were not designed as RDF validation languages. Their use of Open World and Non-Unique Name Assumptions can pose some difficulties to define the integrity constraints that RDF validation languages require [123].

FHIR approach

Another approach for transformation between schemas is to take a domain model as the main representation of data structure and constraints and then transform between that model and other schema formats like XML Schema, JSON Schema or ShEx. This has been the approach followed by FHIR[1]. However, this technique needs the creation of a domain model as an abstract representation which is not the goal of our work.

RDF validation languages and its conversions

Various languages have recently been developed for RDF validation. Shapes Constraint Language (SHACL) [69] has been developed by the W3C Data Shapes Working Group and Shape Expressions (ShEx) [106] is being developed by the W3C Shape Expressions Community Group.

To the best of our knowledge, no conversion between XML Schema and ShEx/SHACL has been proposed to date. This might be due to the recent introduction of ShEx and SHACL.

In this paper, ShEx is used to describe the mappings due to its compact syntax and its support for recursion whereas in SHACL recursion depends on the implementation. However, we consider that converting the mappings proposed in this paper to SHACL is feasible and can be an interesting line of future work given that it has already been accepted as a W3C recommendation and that there are some ways to simulate recursion by target declarations or property paths.

3.3 Brief introduction to ShEx

ShEx was proposed as a language for RDF validation in 2014 [106]. It was one of the foundations for the W3C Data Shapes Working Group which developed the Shapes Constraint Language (SHACL) for the same purpose. SHACL was also inspired by SPIN [68] and although both languages can perform RDF validation there are some differences between them like the support of recursion or the emphasis on validation versus constraint checking (see chapter 7 of [72]

[1] `https://www.hl7.org/fhir/`

for more details). In this paper, we will focus on ShEx because it has a well-defined semantics for recursion [17] and its semantics are more inspired by grammar-based formalisms like Relax NG.

ShEx syntax was inspired by Turtle, SPARQL and Relax NG with the aim to offer a concise and easy to use syntax. In July 2017, version 2.0 was released together with a draft community group report and the community group is currently developing version 2.1.

ShEx uses shapes to group different validations associated with the same node 'type'. That is, a shape can define how a node and its triples should be in order to be valid. Listing 3.1 illustrates an example of a ShEx document defining a shape with a `:PurchaseOrder` type.

```
PREFIX : <http://example.com/>
PREFIX schema: <http://schema.org>
PREFIX
    xs: <http://www.w3.org/2001/XMLSchema#>

:PurchaseOrder {
  :orderId           /Order\\d{2}/ ;
  schema:customer    @:User ;
  schema:orderDate   xs:date ? ;
  schema:orderedItem  @:Item +
}
:Item {
  schema:name xs:string ;
  :quantity xs:positiveInteger OR
            xs:integer MININCLUSIVE 1
}
:User {
  a [ schema:Person ] ;
  :purchaseOrder @:PurchaseOrder*
}
```

Listing 3.1: ShEx shape example

Prefixes are defined at the beginning of the snippet and use the same syntax as in Turtle. Triple constraints are defined inside the shape where a purchase order must have an `orderId` value that matches the regular expression `Order\d{2}`, it must have a `schema:customer` value which must be a node that conforms to shape `:User`, a `schema:orderDate` whose value must be of type `xs:date` and can have one or more (represented by the plus sign) `schema:orderedItem` whose values must conform to the `:Item` shape.

The `:Item` shape must have a `schema:name` value of type `xs:string` and a `:quantity` value of type `xs:positiveInteger`, while the `:User` shape declares that the values must have type `schema:Person`, and can contain zero or more values of `:purchaseOrder` which must conform to the `:PurchaseOrder` shape.

```
### Pass validation as :PurchaseOrder
:order1 :orderId        "Order23" ;
 schema:customer        :alice ;
 schema:orderDate       "2017-03-02"^^xs:date;
 schema:orderedItem     :item1 .
:alice a                schema:Person ;
       :purchaseOrder   :order1 .
:item1 schema:name      "Lawn" ;
      :quantity         2 .

### Fails validation as :PurchaseOrder
```

```
:order2 :orderId       "MyOrder" ;
 schema:customer     :bob;
 schema:orderDate    2017;
 schema:orderedItem :item1 .
:bob a                 schema:Person ;
    :purchaseOrder   :unknown.
```

Listing 3.2: RDF validation example

In Listing 3.2 there is an example of two purchase orders defined in RDF. The first one passes validation and conforms to the shapes declaration given in Listing 3.1 whereas `:order2` fails validation for several reasons: the value of `:orderId` does not conform to the required regular expression, the value of `schema:customer` does not conform to shape `:User` and the value of `schema:orderDate` does not have datatype `xs:date`.

ShEx supports different serialization formats:

- ShExC: a concise human readable compact syntax which is the one presented in previous example.

- ShExJ: a JSON-LD syntax which is used as an abstract syntax in the ShEx specification [104].

- ShExR: an RDF representation syntax based on ShExJ.

ShEx defines an extension mechanism through which users can embed portions of code written in a programming language or SPARQL. This feature is known as Semantic Actions and are introduced between definition of triples with the *%interpreter{}%* syntax where interpreter is the name of the interpreter to be used (e.g., JS, SPARQL, JAVA). See Listings 3.22 and 3.23 for Semantic Actions examples.

In this paper, ShExC syntax was used because it is easy to read and understand. The goal of this introduction was to provide a basic understanding of ShEx. For more examples and a longer comparison between ShEx and SHACL readers can consult [72].

3.4 Mappings between XML Schema and ShEx

XML Schema defines a set of elements and datatypes for validation that need to be converted to ShEx. In this section, we describe different XML Schema elements and a possible conversion to ShEx. All examples use the default prefix : for URIs. It is intended to be replaced by different prefixes depending on the required namespaces. For XML Schema elements and datatypes `xs` prefix is used in the examples.

Element

Elements are treated as a triple predicate and object, i.e., we convert them to a triple constraint whose predicate is the name of the element:

```
### XML Schema
<xs:element name="birthday" type="xs:date"/>

### ShEx
```

```
:birthday xs:date ;
```

Listing 3.3: Element mapping

The `name` attribute is used as the fragment of the URI in the predicate and the type is transcribed directly, as ShEx has built-in support for XML Schema datatypes. If the `ref` attribute is present, the type must be defined somewhere in the document to link the corresponding type or shape. When an `xs:element` type is a `xs:complexType`, the type should be referenced to a new shape where the `xs:complexType` is converted (see Section 3.4 where we explain how to convert `xs:complexType` to a shape). See Listings 3.3, 3.4 and 3.5 for a list of examples on how to convert an element.

```
### XML Schema
<xs:element name="purchaseOrder"
            type="PurchaseOrderType"/>

<xs:complexType name="PurchaseOrderType">
    ...
</xs:complexType>

### ShEX
:purchaseOrder @<PurchaseOrderType> ;
```

Listing 3.4: Element mapping with linked type

```
### XML Schema
<xs:element name="item"
            minOccurs="0"
            maxOccurs="unbounded">
    <xs:complexType>
        ...
    </xs:complexType>
</xs:element>

### ShEx
:item @<item> * ;
```

Listing 3.5: Element mapping with nested type

As presented in Listing 3.5, when an element has its complex type nested the shape name will be the `name` of the element.

Cardinality

Cardinality in ShEx is defined with the following symbols: '*' for 0 or more repetitions, '+' for 1 or more repetitions, '?' for 0 or 1 repetitions (optional element) or '{m, n}' for m to n repetitions where m is `minOccurs` and n `maxOccurs`. As in XML Schema, the default cardinality in ShEx is 1 for lower and upper bounds. Therefore, transformation of `minOccurs` and `maxOccurs` in the previously defined cardinality marks is done as showed in Listing 3.6.

```
### XML Schema
<xs:element name="nameZeroUnbounded"
            type="xs:string"
            minOccurs="0"
            maxOccurs="unbounded">
<xs:element name="nameOneUnbounded"
```

```
            type="xs:string"
            minOccurs="1"
            maxOccurs="unbounded">
<xs:element name="nameOptional"
            type="xs:string"
            minOccurs="0"
            maxOccurs="1">
<xs:element name="nameFourToTen"
            type="xs:string"
            minOccurs="4"
            maxOccurs="10">

### ShEx
:nameZeroUnbounded xs:string * ;
:nameOneUnbounded xs:string + ;
:nameOptional xs:string ? ;
:nameFourToTen xs:string {4, 10} ;
```

Listing 3.6: Cardinality mapping

Attribute

ShEx treats attributes like elements because it makes no difference between an attribute and an element. This difference is part of XML data model whereas the RDF data model does not have the concept of attributes. One possibility to transform attributes is to use their `name` and `type` as performed with elements (see Section 3.4). This allows better readability of the corresponding RDF data, but limits roundtrip conversions between XML to RDF and back.

ComplexType

Complex types are translated directly to ShEx shapes. The `name` of the `xs:complexType` will be the name of the shape to which elements can refer to (see Listing 3.7 for an example). Complex types consist of various statements, so we provide a detailed transformation of each possibility in the following sections.

```
### XML Schema
<xs:complexType name="PurchaseOrderType">
    ...
</xs:complexType>

### ShEx
<PurchaseOrderType> {
    ...
}
```

Listing 3.7: Complex type mapping

Sequence

While sequences in XML Schema define sequential order of elements, representing the same modeling in ShEx is complex due to RDF graph structure. There are several ways to represent order in RDF, the most obvious one is using RDF lists (cf., other ways to represent it [38, 86]).

The example in Listing 3.8 shows how the mapping is done for a `xs:sequence` using RDF lists:

```
### XML Schema
<xs:complexType name="Address">
    <xs:sequence>
        <xs:element name="street"
            type="xs:string"/>
        <xs:element name="city"
            type="xs:string"/>
        <xs:element name="state"
            type="xs:string"/>
        <xs:element name="zip"
            type="xs:decimal"/>
    </xs:sequence>
</xs:complexType>

### ShEx
<address> {
    rdf:first @<street> ;
    rdf:rest @<i1> ;
}
<i1> {
    rdf:first @<city> ;
    rdf:rest @<i2> ;
}
<i2> {
    rdf:first @<state> ;
    rdf:rest @<i3> ;
}
<i3> {
    rdf:first @<zip> ;
    rdf:rest [ rdf:nil ] ;
}
<street> {
    :street xs:string ;
}
<city> {
    :city xs:string ;
}
<state> {
    :state xs:string ;
}
<zip> {
    :zip xs:decimal ;
}
```

Listing 3.8: Sequence mapping

Choice

Choices in XML Schema are the disjunction operator to select between two options, for instance: choice between two elements. This operator is supported in ShEx using the *oneOf* operator ('—'). The object and predicate of the RDF statement must be one of the enclosed ones. Therefore, translation is performed as shown in the snippet of Listing 3.9:

```
### XML Schema
<xs:choice>
  <xs:element name="name"
              type="xs:string"/>
  <xs:all>
    <xs:element name="givenName"
```

```
                type="xs:string"
                maxOccurs="unbounded"/>
    <xs:element name="familyName"
                type="xs:string" />
  </xs:all>
</xs:choice>

### ShEx
( :name xs:string |
  :givenName xs:string + ;
  :familyName xs:string
) ;
```

Listing 3.9: Choice mapping

All

While sequences are an ordered set of elements, `xs:all` is instead a set of unordered elements. Indeed, `xs:all` has a better representation using ShEx elements and the transformation is simpler than the `xs:sequence` one as there is no need to keep track of the order of elements. See Listing 3.10 for an example.

```
### XML Schema
<xs:all>
    <xs:element name="street"
                type="xs:string"/>
    <xs:element name="city"
                type="xs:string"/>
    <xs:element name="state"
                type="xs:string"/>
    <xs:element name="zip"
                type="xs:decimal"/>
</xs:all>

### ShEx
:street  xs:string  ;
:city    xs:string  ;
:state   xs:string  ;
:zip     xs:decimal ;
```

Listing 3.10: All mapping

XSD Types

XSD Types can be used in ShEx as they are used on XML Schema, e.g., whenever a string type is required we can use `xs:string`. Therefore, translation is done directly using the same types that are defined in the XML Schema document.

Enumerations (using NMTokens)

Enumerations in XML Schema can be used to declare the possible values that an element can have. In ShEx, this is supported using the symbols '[' and ']'. The enclosed values are the possible values that the RDF object can take. See Listing 3.11 for an example.

```
### XML Schema
<xs:simpleType name="PublicationType">
   <xs:restriction base="xs:NMTOKEN">
     <xs:enumeration value="Book"/>
     <xs:enumeration value="Magazine"/>
     <xs:enumeration value="Journal"/>
  </xs:restriction>
</xs:simpleType>

<xs:element name="pubType"
            ref="PublicationType"/>
<xs:attribute name="country"
              type="xs:NMTOKEN"
              fixed="US"/>

### ShEx
:pubType ["Book" "Magazine" "Journal"] ;
:country ["US"] ;
```

Listing 3.11: Enumarations (using NMTokens) mapping

Pattern

xs:pattern is used in XML Schema to define the format and allowed contents of a string value. xs:pattern in ShEx uses a syntax similar to the JavaScript language except that backslash is required to be escaped, i.e., double backslash has to be used to correctly escape. Therefore, the conversion is a transformation between XML Schema and JavaScript Regular Expression syntaxes as shown in Listing 3.12.

```
### XML Schema
<xs:simpleType name="SKU">
  <xs:restriction base="xs:string">
    <xs:pattern value="\d{3}-[A-Z]{2}"/>
  </xs:restriction>
</xs:simpleType>
<xs:attribute name="partNum"
              type="SKU"
              use="required"/>

### ShEx
:partNum /\\d{3}-[A-Z]{2}/ ;
```

Listing 3.12: Pattern mapping

SimpleType

Simple types in XML Schema are based on XSD Types (see Section 3.4) and allow some enhancements like: restrictions, lists and unions. Depending on the content, translation is performed following different strategies which we detail bellow. For translation of restrictions, see Section 3.4.

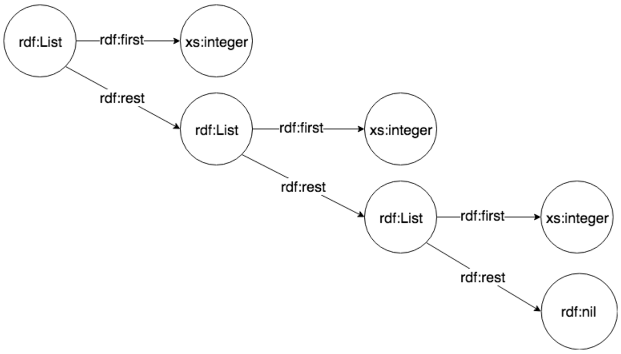


Figure 3.1: Example of a RDF list construction

List

Lists inside simple types define a way of creating collections of a base XSD type in XML Schema. These lists are supported in RDF using RDF Collections[2]. As previously discussed, there can be several approaches to represent ordered lists in RDF (see Section 3.4). A commonly accepted approach is the use of RDF lists: the `rdf:first` edge points to the first element and the `rdf:rest` edge to the rest of the list which recursively follows the same structure until the `rdf:nil` element is declared to represent the end of the list. This way, it is possible to create the desired list and preserve the order. Figure 3.1 shows how an RDF list is constructed for a better understanding of this section. Hence, translation into ShEx is made by using RDF lists and the use of recursion that defines a type with a pointer to itself in the `rdf:rest` edge. See Listing 3.13 for an example.

```
### XML Schema
<xs:simpleType name="IntegerList">
  <xs:list itemType="xs:integer" />
</xs:simpleType>

### ShEx
<IntegerList> {
 rdf:first xs:integer ;
 rdf:rest @<IntegerList> OR [rdf:nil];
}
```

Listing 3.13: List mapping

Union

Unions are the mechanism that XML Schema offers to make new types that are the combination of two simple types. With this kind of disjunction, a new type which allows any value admitted by any of the members of the `xs:union` is created. For the translation into ShEx we create a new type that is the combination of the types involved in the `xs:union` as shown in Listing 3.14.

```
### XML Schema
```

[2] https://www.w3.org/TR/rdf11-mt/\#rdf-collections

```
<xs:attribute name="fontsize">
 <xs:simpleType>
   <xs:union memberTypes="Fontbynumber
                          Fontbystringname"
   />
    </xs:simpleType>
</xs:attribute>

<xs:simpleType name="Fontbynumber">
    <xs:restriction
        base="xs:positiveInteger">
        <xs:maxInclusive value="72"/>
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="Fontbystringname">
    <xs:restriction base="xs:string">
        <xs:enumeration value="small"/>
        <xs:enumeration value="medium"/>
        <xs:enumeration value="large"/>
    </xs:restriction>
</xs:simpleType>

### ShEx
:fontsize
  @<Fontbynumber> OR @<Fontbystringname>

<Fontbynumber>
   xs:positiveInteger MAXINCLUSIVE 72

<Fontbystringname> ["small"
                   "medium"
                   "large"
                  ]
```

Listing 3.14: Union mapping

Complex Content and Simple Content

Complex contents and simple contents are a way to define a new type from a base type using restrictions or extensions. The base type is the one that is used as a base for the restriction (or extension) clause and the new type is the one that is been restricted (or extended). Complex content allows to extend or restrict a base `xs:complexType` with mixed content or elements only. Simple content allows to extend or restrict a `xs:complexType` with character data or with a `xs:simpleType`. For the translation into ShEx, the respective `xs:restriction` or `xs:extension` have to be taken into account to define the new type.

Restriction

Restrictions are used in XML Schema to restrict possible values of a base type. A new type can be defined using restrictions applied to a base type. Depending on how the type and the restrictions are defined, the translation strategies vary.

- Simple Content: If `xs:simpleContent` is present, XSD Facets/Restrictions must be used (see Section 3.4 for more information). When restricting using a `xs:simpleType`, the transformation is done using the known base type (see Section 3.4) and putting some format restrictions to it. Translation into ShEx will be performed using the base type and translating the XSD Facets as they are defined in every specific case (see Section 3.4).

- Complex Content: If `xs:complexContent` is present, the base `xs:complexType` is restricted using `xs:all`, `xs:group`, `xs:choice`, `xs:sequence`, `xs:attribute` or `xs:attributeGroup`. Complex content restriction will restrict allowed values and elements types. This is a case of inheritance by restriction. For translation into ShEx, the `xs:restriction` elements must be taken and transformed directly into a new shape that defines the resulting child shape[3].

Extension

With extensions in XML Schema, it is possible to define a new type as an extension of a previously defined one. This is a case of classic inheritance, where the child inherits its parent elements that are added to its own defined elements. Depending on the content, i.e., `xs:complexContent` or `xs:simpleContent`, different translation strategies can be used.

- Simple content: If `xs:simpleContent` is present, extension of the base type is performed by adding more attributes or attribute groups to the new type. Therefore, the translation into ShEx is made by the concatenation of both the type and its `xs:extension` to create the new shape.

- Complex content: If `xs:complexContent` is present, extension of base type is performed by adding more attributes and elements to a new base one. Therefore, translation is done by combining the base type and its `xs:extension` to create a new shape.

Restrictions and extensions in ShEx are not supported directly in the current version (i.e., ShEx has no support for extensions, restriction or inheritance) with the same semantics as XML Schema. Therefore, we use the normal syntax provided by ShEx and create the two resulting shapes—by solving the `xs:restriction` or `xs:extension` before the translation to ShEx—from the respective `xs:restriction` or `xs:extension` as can be seen in Listing 3.15. However, this translation suffers from a loss of semantics—which is in line with RQ3—which makes impossible a backwards conversion.

```
### XML Schema
<xs:simpleType name="mountainBikeSize">
  <xs:restriction base="xs:string">
    <xs:enumeration value="small" />
    <xs:enumeration value="medium" />
    <xs:enumeration value="large" />
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="FamilyMountainBikes">
  <xs:simpleContent>
    <xs:extension base="mountainBikeSize">
```

[3] Future versions of ShEx are planning to include inheritance. See: https://github.com/shexSpec/shex/issues/50

```
      <xs:attribute name="familyMember">
       <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="child" />
          <xs:enumeration value="male" />
          <xs:enumeration value="female" />
        </xs:restriction>
       </xs:simpleType>
      </xs:attribute>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

### ShEx
<MountainBikeSize> ["small" "medium" "large"]

<FamilyMountainBikes> {
 :mountainBikeSize @<MountainBikeSize> ;
 :familyMember ["child" "male" "female"];
}
```

Listing 3.15: Restrictions and extensions mapping, where extensions and restrictions are directly transformed into the equivalent shape

XSD Types Restrictions/Facets

Enumeration

`xs:enumeration` restriction uses a base type to restrict the possible values of a type. It is declared using a set of possible values. In ShEx, this is defined using the '[' and ']' operators. The values that are allowed are enclosed inside the square brackets. This is the same mechanism how the example in Section 3.4 works. However, Listing 3.16 shows a more complex example using extensions and restrictions.

```
### XML Schema
<xs:simpleType name="Mountainbikesize">
  <xs:restriction base="xs:string">
    <xs:enumeration value="small"/>
    <xs:enumeration value="medium"/>
    <xs:enumeration value="large"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType
  name="FamilyMountainBikeSizes">
  <xs:simpleContent>
    <xs:extension base="mountainbikesize">
       <xs:attribute name="familyMember"
                     type="xs:string" />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>

<xs:complexType
    name="ChildMountainBikeSizes">
  <xs:simpleContent>
     <xs:restriction
          base="FamilyMountainBikeSizes" >
        <xs:enumeration value="small"/>
        <xs:enumeration value="medium"/>
     </xs:restriction>
  </xs:simpleContent>
</xs:complexType>

### ShEx
<MountainBikeSize> ["small" "medium" "large"]

<FamilyMountainBikes> {
 :mountainBikeSize @<MountainBikeSize> ;
 :familyMember ["child" "male" "female"];
}

<ChildMountainBikeSizes>
   @<FamilyMountainBikes> AND {
   :mountainBikeSize ["small" "medium"]
}
```

Listing 3.16: Enumeration mapping

Fraction digits

xs:fractionDigits are used in XML Schema when a decimal type is defined (e.g., xs:decimal) and the number of decimal digits is desired to be restricted in the representation. ShEx supports this feature in a similar way as XML Schema. Hence, FRACTIONDIGITS keyword is used followed by the integer number of fraction digits that should be allowed. See Listing 3.17 for an example.

```
### XML Schema
<xs:element name="itemValue">
  <xs:simpleType>
    <xs:restriction base="xs:decimal">
      <xs:fractionDigits value="2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

### ShEx
:itemValue xs:decimal FRACTIONDIGITS 2 ;
```

Listing 3.17: Fraction digits mapping

Total digits

This feature allows to restrict the total number of digits permitted in a numeric type. In ShEx, this is possible using TOTALDIGITS keyword as shown in Listing 3.18.

```
### XML Schema
<xs:element name="age">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:totalDigits value="3"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

### ShEx
:age xs:integer
```

```
     TOTALDIGITS 3 ;
```

Listing 3.18: Total digits mapping

Length

xs:length is used to restrict the number of characters allowed in a string type. In ShEx, this is supported with the LENGTH keyword followed by the integer number that defines the desired length as shown in Listing 3.19.

```
### XML Schema
<xs:element name="group">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:length value="1"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

### ShEx
:group xs:string LENGTH 1 ;
```

Listing 3.19: Length mapping

Max Length and Min Length

xs:maxLength and xs:minLength are used to restrict the number of characters allowed in a text type. But instead of restricting to a fixed number of characters, with these features restriction to a length interval is possible. In ShEx, the definitions of minimum and maximum length are made by using the MINLENGTH and MAXLENGTH keywords as shown in Listing 3.20.

```
### XML Schema
<xs:element name="comments">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:minLength value="1"/>
      <xs:maxLength value="1000"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

### ShEx
:comment xs:string
         MINLENGTH 1
         MAXLENGTH 1000;
```

Listing 3.20: Max length and min length mapping

Max-min exclusive and max-min inclusive

These features allow restricting number types to an interval of desired values. This is the same notion as in open and closed intervals. In ShEx, these features are supported directly. Therefore, transformation is done as shown in Listing 3.21.

```
### XML Schema
<xs:element name="cores">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minExclusive value="0"/>
      <xs:maxExclusive value="9"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="coresOpenInterval">
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="8"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

### ShEx
:cores xs:integer
       MINEXCLUSIVE 0
       MAXEXCLUSIVE 9 ;
:coresOpenInterval xs:integer
                   MININCLUSIVE 1
                   MAXINCLUSIVE 8 ;
```

Listing 3.21: Max exclusive, min exclusive, min inclusive and max inclusive mapping

Whitespace

`xs:whiteSpace` allows to specify how white spaces in strings are handled. In XML Schema, there are three options:

- Preserve: This option will not remove any white space character from the given string.
- Replace: This option will replace all white space characters (line feeds, tabs, spaces and carriage returns) with spaces.
- Collapse: This option will remove all white spaces characters:
  - Line feeds, tabs, spaces and carriage returns are replaced with spaces.
  - Leading and trailing spaces are removed.
  - Multiple spaces are reduced to a single space.

In ShEx, `xs:whiteSpace` options are not supported. Their behaviour could be simulated using semantic actions (see Listing 3.22).

```
### XML Schema
<xs:complexType name="whiteSpaces">
 <xs:all>
  <xs:element name="preserve">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:whiteSpace
           value="preserve"/>
        </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="replace">
    <xs:simpleType>
       <xs:restriction base="xs:string">
          <xs:whiteSpace
             value="replace"/>
          </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="collapse">
   <xs:simpleType>
     <xs:restriction base="xs:string">
        <xs:whiteSpace
           value="collapse"/>
     </xs:restriction>
   </xs:simpleType>
 </xs:element>
</xs:all>
</xs:complexType>

### ShEx
<whiteSpaces> {
  :preserve xs:string ;
  :replace  xs:string
   %js{
     _.o.lex = _.o.lex
        .replace("/\r|\n|\r\n|\s/g", " ");
    return true;
   }
   % ;
   :collapse xs:string
   %js{
     var replacedText = _.o.lex
        .replace("/\r|\n|\r\n|\s/g", " ");
     _.o.lex = replacedText.trim();
     return true;
   }
  %
}
```

Listing 3.22: WhiteSpace mapping

Unique

xs:unique is used in XML Schema to define that an element of some type is unique, i.e., there cannot be the same values among elements defined in the rule. This is useful for cases like IDs, where a unique ID is the way to identify an element. Currently, ShEx does not support Unique function but it is expected to be supported in future versions[4]. As a temporal solution, semantic actions could be used to implement this kind of constraint (see Listing 3.23).

```
### XML Schema
<xs:element name="Person"
            maxOccurs="unbounded">
  <xs:complexType>
    <xs:all>
       <xs:element name="name"
```

[4] https://www.w3.org/2001/sw/wiki/ShEx/Unique_UNIQUE

```
                   type="xs:string" />
         <xs:element name="surname"
                     type="xs:string" />
           <xs:element name="id"
                       type="xs:integer" />
    </xs:all>
  </xs:complexType>
  <xs:unique name="onePersonPerID">
     <xs:selector xpath="."/>
     <xs:field xpath="id"/>
  </xs:unique>
</xs:element>

### ShEx
%js{
    var ids = [];
    return true;
}
%
<Person> {
  :name xs:string ;
  :surname xs:string ;
  :id xs:integer
  %js{ if(ids.indexOf(_.o.lex) >= 0)
         return false;
       ids.push(_.o.lex);
       return true;
  }%
}
```

Listing 3.23: Unique mapping

3.5 XMLSchema2ShEx prototype

In addition to the proposed mappings from XML Schema to Shape Expressions, and in order to answer RQ2, a prototype has been developed. This prototype uses a subset of the presented mappings and converts a given XML Schema input to a ShEx output.

The prototype has been developed in Scala and is available online[5]. It is a work-in-progress implementation, so not all the mappings are supported yet (see Table 3.1 for a list of supported features).

The tool is built on top of Scala parser combinators [94]. Once the XML Schema input is analysed and verified, it is converted to ShEx based on different elements and types declared on it. These conversions are made recursively and printed to the ouput in ShEx Compact Format (ShExC).

The input XML Schema document example presented in Listing 3.24 is used to ensure that the prototype can work and do the transformation as expected. This example includes complex types, attributes, elements, simple types and patterns among others. Complex types are converted to shapes, elements and attributes to triple predicates and objects, restrictions (max/minExclusive and max/minInclusive) to numeric intervals, cardinality attributes to ShEx cardinality and so on. Although it is a small example, it has the structure of typical XML Schemas used nowadays and the prototype can convert it properly as it is stated in Listing 3.24.

[5] https://github.com/herminiogg/XMLSchema2ShEx

Table 3.1: Supported and pending of implementation features in XMLSchema2ShEx prototype. * Not natively supported in ShEx 2.0.

Supported features	Complex type, Simple type, All, Attributes, Restriction, Element, Max exclusive, Min exclusive, Max inclusive, Min inclusive, Enumeration, Pattern, Cardinality
Pending implementation	Choice, List, Union, Extension, Fraction Digits, Length, Max Length, Min Length, Total digits, Whitespace*, Unique*

```
### XML Schema
<xs:schema
 xmlns:xs="http://www.w3.org/2001/XMLSchema"
 targetNamespace="http://tempuri.org/po.xsd"
 xmlns="http://tempuri.org/po.xsd"
 elementFormDefault="qualified">

 <xs:element name="purchaseOrder"
             type="PurchaseOrderType"/>

  <xs:element name="comment"
              type="xs:string"/>

  <xs:complexType name="PurchaseOrderType">
    <xs:all>
       <xs:element name="shipTo"
                   type="USAddress"/>
       <xs:element name="billTo"
                   type="USAddress"/>
       <xs:element ref="comment"
                   minOccurs="0"/>
       <xs:element name="items"
                   type="Items"/>
    </xs:all>
    <xs:attribute name="orderDate"
                  type="xs:date"/>
  </xs:complexType>

  <xs:complexType name="USAddress">
    <xs:all>
        <xs:element name="name"
                    type="xs:string"/>
        <xs:element name="street"
                    type="xs:string"/>
        <xs:element name="city"
                    type="xs:string"/>
        <xs:element name="state"
                    type="xs:string"/>
        <xs:element name="zip"
                    type="xs:integer"/>
    </xs:all>
    <xs:attribute name="country"
```

```
                  type="xs:NMTOKEN"
                  fixed="US"/>
 </xs:complexType>

 <xs:complexType name="Items">
   <xs:all>
    <xs:element name="item"
                minOccurs="0"
                maxOccurs="unbounded">
     <xs:complexType>
      <xs:all>
       <xs:element
                name="productName"
                type="xs:string"/>
       <xs:element
                name="quantity">
         <xs:simpleType>
            <xs:restriction
               base="xs:positiveInteger">
              <xs:maxExclusive
                   value="100"/>
            </xs:restriction>
         </xs:simpleType>
         </xs:element>
           <xs:element name="USPrice"
              type="xs:decimal"/>
         <xs:element ref="comment"
              minOccurs="0"/>
         <xs:element name="shipDate"
              type="xs:date" minOccurs="0"/>
     </xs:all>
   <xs:attribute name="partNum" type="SKU"
        use="required"/>
   </xs:complexType>
   </xs:element>
  </xs:all>
 </xs:complexType>

 <xs:simpleType name="SKU">
   <xs:restriction base="xs:string">
     <xs:pattern value="\d{3}-[A-Z]{2}"/>
   </xs:restriction>
 </xs:simpleType>
</xs:schema>

### ShEx
PREFIX : <http://www.example.com/>
PREFIX
    xs: <http://www.w3.org/2001/XMLSchema#>

<Items> {
  :item         @<item> *  ;
}
<item> {
  :productName  xs:string  ;
  :quantity     xs:positiveInteger
                MAXEXCLUSIVE 100 ;
  :USPrice      xs:decimal  ;
  :comment      xs:string ?  ;
  :shipDate     xs:date ?  ;
  :partNum      /\\d{3}-[A-Z]{2}/ ;
}
```

```
<PurchaseOrderType> {
  :shipTo     @<USAddress>  ;
  :billTo     @<USAddress> ;
  :comment    xs:string ?  ;
  :items      @<Items>  ;
  :orderDate  xs:date ;
}
<USAddress> {
  :name       xs:string ;
  :street     xs:string  ;
  :city       xs:string  ;
  :state      xs:string ;
  :zip        xs:integer ;
  :country    ["US"]  ;
}
```

Listing 3.24: XML Schema to ShEx example

Validation example

```
### XML
<?xml version="1.0"?>
<purchaseOrder
  xmlns="http://tempuri.org/po.xsd"
  orderDate="1999-10-20">
  <shipTo country="US">
     <name>Alice Smith</name>
     <street>123 Maple Street</street>
     <city>Mill Valley</city>
     <state>CA</state>
     <zip>90952</zip>
  </shipTo>
  <billTo country="US">
     <name>Robert Smith</name>
     <street>8 Oak Avenue</street>
     <city>Old Town</city>
     <state>PA</state>
     <zip>95819</zip>
  </billTo>
  <comment>
     Hurry, my lawn is going wild!
  </comment>
  <items>
  <item partNum="872-AA">
     <productName>
       Lawnmower
     </productName>
     <quantity>1</quantity>
     <USPrice>148.95</USPrice>
      <comment>
                Confirm this is electric
      </comment>
      </item>
  <item partNum="926-AA">
    <productName>
        Baby Monitor
    </productName>
    <quantity>1</quantity>
    <USPrice>39.98</USPrice>
    <shipDate>1999-05-21</shipDate>
  </item>
```

```
 </items>
</purchaseOrder>

### RDF
:order1
  :shipTo [
    :name "Alice Smith" ;
    :street "123 Maple Street" ;
    :city "Mall Valley" ;
    :state "CA" ;
    :zip 90952 ;
    :country "US"
  ] ;
 :billTo [
    :name "Robert Smith" ;
    :street "8 Oak Avenue" ;
    :city "Old Town" ;
    :state "PA" ;
    :zip 95819 ;
    :country "US"
  ] ;
 :comment "Hurry, my lawn is going wild!";
 :items [
    :item [
        :productName "Lawnmower" ;
        :quantity "1"^^xs:positiveInteger ;
        :USPrice 148.95 ;
        :comment "Confirm this is electric";
        :partNum "872-AA"
    ] ;
    :item [
        :productName "Baby Monitor" ;
        :quantity "1"^^xs:positiveInteger ;
        :USPrice 39.98 ;
        :shipDate "1999-05-21"^^xs:date ;
        :partNum "926-AA"
    ] ;
 ];
 :orderDate "1999-10-20"^^xs:date .
```

Listing 3.25: XML to RDF example

Once conversion from XML Schema to ShEx is done, it must be verified that the same validation that was performed on XML data using XML Schema, but now on RDF data using ShEx, is working equivalently. The translation of a valid XML to RDF is executed which is presented in Listing 3.25. The conversion presented in the snippet uses blank nodes to represent the nested types. This is done to avoid creating a fictitious node every time a triple is pointing to another triple (in other words, every time it has a nested type). The conversion was performed following similar equivalences to those proposed in the mappings. That is, complex types to triple subjects or predicates, simple types to triple objects, cardinality translated directly and so on.

For RDF validation using ShEx there are various implementations in different programming languages that are being developed[6]. One of these implementations is made in Scala by one of the authors of this paper and it is available online[7].

[6] A list of ShEx implementations is available at: https://shex.io

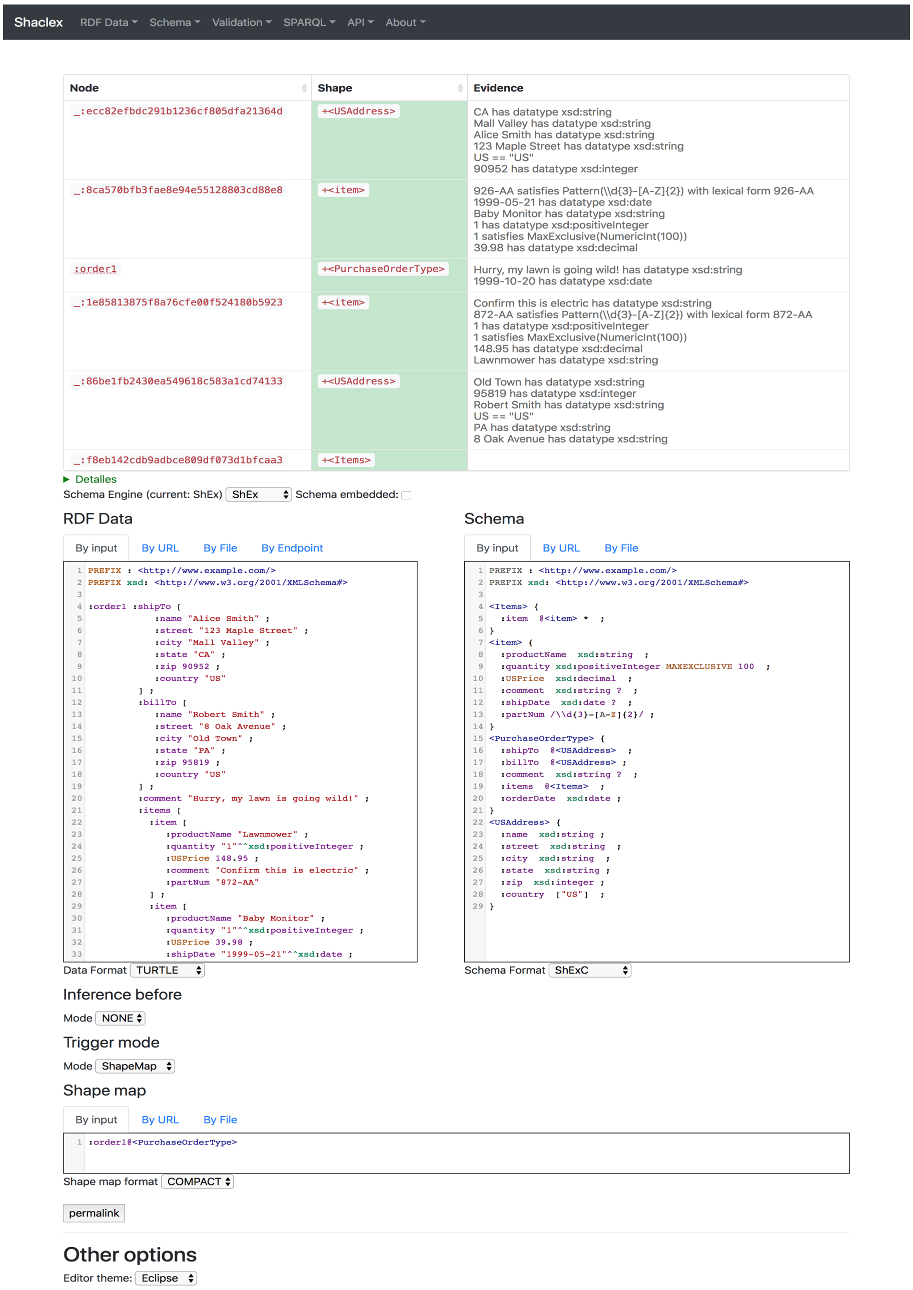
[7] http://shaclex.herokuapp.com

Using the examples given above the validation can be performed with the mentioned tool which allows the RDF and the ShEx inputs in various formats and then the option to validate the RDF against ShEx or SHACL schema. As seen in Figure 3.2, validation is performed trying to match the shapes with the existing graphs, whenever the tool matches a pattern it shows the evidence in green and a short explanation of why this graph has matched.

3.6 Non-Deterministic schemata

There is an issue that arises in XML Schema documents that should be solved when proposing a transformation from XML Schema. This is the topic of Non-Deterministic schemata where the parser is unable to determine the sequence to validate due to the Unique Particle Attribution. This issue appears, for example, in a choice between two sequences that begin with the same element. This event can be formulated with the regular expression: `(ab | ac)` and in XML Schema as shown in Listing 3.26.

```
### XML Schema
<xs:complexType name="nondeterministic">
    <xs:choice>
        <xs:sequence>
            <xs:element name="a"/>
            <xs:element name="b"/>
        </xs:sequence>
        <xs:sequence>
            <xs:element name="a"/>
            <xs:element name="c"/>
        </xs:sequence>
    </xs:choice>
</xs:complexType>

### ShEx
<nondeterministic> {
  a @<ab> OR @<ac> ;
}

<ab> {
  rdf:first @<a> ;
  rdf:rest @<ab1> ;
}

<ac> {
  rdf:first @<a> ;
  rdf:rest @<ac1> ;
}

<ab1> {
  rdf:first @<b> ;
  rdf:rest [rdf:nil]  ;
}

<ac1> {
  rdf:first @<c> ;
  rdf:rest [rdf:nil]  ;
}

<a> {
  :namea xs:string ;
}
```



Node	Shape	Evidence
_:ecc82efbdc291b1236cf805dfa21364d	+<USAddress>	CA has datatype xsd:string Mall Valley has datatype xsd:string Alice Smith has datatype xsd:string 123 Maple Street has datatype xsd:string US == "US" 90952 has datatype xsd:integer
_:8ca570bfb3fae8e94e55128803cd88e8	+<item>	926-AA satisfies Pattern(\\d{3}-[A-Z]{2}) with lexical form 926-AA 1999-05-21 has datatype xsd:date Baby Monitor has datatype xsd:string 1 has datatype xsd:positiveInteger 1 satisfies MaxExclusive(NumericInt(100)) 39.98 has datatype xsd:decimal
:order1	+<PurchaseOrderType>	Hurry, my lawn is going wild! has datatype xsd:string 1999-10-20 has datatype xsd:date
_:1e85813875f8a76cfe00f524180b5923	+<item>	Confirm this is electric has datatype xsd:string 872-AA satisfies Pattern(\\d{3}-[A-Z]{2}) with lexical form 872-AA 1 has datatype xsd:positiveInteger 1 satisfies MaxExclusive(NumericInt(100)) 148.95 has datatype xsd:decimal Lawnmower has datatype xsd:string
_:86be1fb2430ea549618c583a1cd74133	+<USAddress>	Old Town has datatype xsd:string 95819 has datatype xsd:integer Robert Smith has datatype xsd:string US == "US" PA has datatype xsd:string 8 Oak Avenue has datatype xsd:string
_:f8eb142cdb9adbce809df073d1bfcaa3	+<Items>	

▶ Detalles

Schema Engine (current: ShEx) ShEx Schema embedded: □

RDF Data

By input By URL By File By Endpoint

```
PREFIX : <http://www.example.com/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

:order1 :shipTo [
            :name "Alice Smith" ;
            :street "123 Maple Street" ;
            :city "Mall Valley" ;
            :state "CA" ;
            :zip 90952 ;
            :country "US"
         ] ;
         :billTo [
            :name "Robert Smith" ;
            :street "8 Oak Avenue" ;
            :city "Old Town" ;
            :state "PA" ;
            :zip 95819 ;
            :country "US"
         ] ;
         :comment "Hurry, my lawn is going wild!" ;
         :items [
           :item [
              :productName "Lawnmower" ;
              :quantity "1"^^xsd:positiveInteger ;
              :USPrice 148.95 ;
              :comment "Confirm this is electric" ;
              :partNum "872-AA"
           ] ;
           :item [
              :productName "Baby Monitor" ;
              :quantity "1"^^xsd:positiveInteger ;
              :USPrice 39.98 ;
              :shipDate "1999-05-21"^^xsd:date ;
```

Data Format TURTLE

Schema

By input By URL By File

```
PREFIX : <http://www.example.com/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

<Items> {
  :item  @<item> *  ;
}
<item> {
  :productName  xsd:string  ;
  :quantity xsd:positiveInteger MAXEXCLUSIVE 100  ;
  :USPrice  xsd:decimal  ;
  :comment  xsd:string ?  ;
  :shipDate  xsd:date ?  ;
  :partNum /\\d{3}-[A-Z]{2}/ ;
}
<PurchaseOrderType> {
  :shipTo  @<USAddress>  ;
  :billTo  @<USAddress> ;
  :comment  xsd:string ?  ;
  :items  @<Items>  ;
  :orderDate  xsd:date ;
}
<USAddress> {
  :name  xsd:string ;
  :street  xsd:string  ;
  :city  xsd:string  ;
  :state  xsd:string ;
  :zip  xsd:integer ;
  :country  ["US"]  ;
}
```

Schema Format ShExC

Inference before

Mode NONE

Trigger mode

Mode ShapeMap

Shape map

By input By URL By File

```
:order1@<PurchaseOrderType>
```

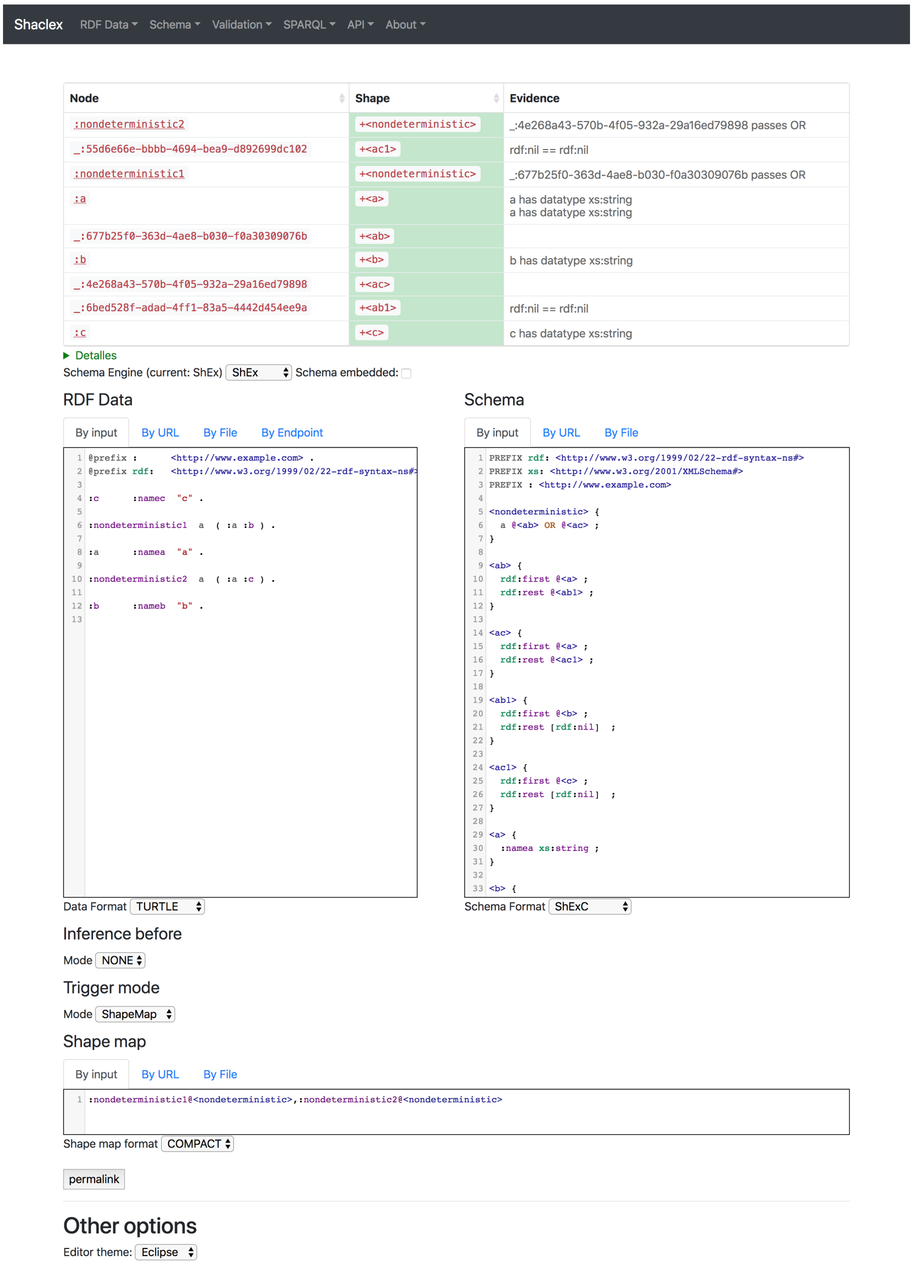
Shape map format COMPACT

permalink

Other options

Editor theme: Eclipse

Figure 3.2: Validation result using Shaclex validator. The RDF data is entered in the left text area whereas the ShEx schema is entered on the right text area. In the bottom, a ShapeMap is declared to make the validator know where and how to begin the validation, in this case we commanded to validate :order1 node with ¡PurchaseOrderType¿ shape. In the top of the page, the result is shown detailing how each node was validated and what are the evidences or failures for the validation. A link to the validation example can be found in Supplementary Material.



Node	Shape	Evidence
:nondeterministic2	+<nondeterministic>	_:4e268a43-570b-4f05-932a-29a16ed79898 passes OR
_:55d6e66e-bbbb-4694-bea9-d892699dc102	+<ac1>	rdf:nil == rdf:nil
:nondeterministic1	+<nondeterministic>	_:677b25f0-363d-4ae8-b030-f0a30309076b passes OR
:a	+<a>	a has datatype xs:string a has datatype xs:string
_:677b25f0-363d-4ae8-b030-f0a30309076b	+<ab>	
:b	+<b>	b has datatype xs:string
_:4e268a43-570b-4f05-932a-29a16ed79898	+<ac>	
_:6bed528f-adad-4ff1-83a5-4442d454ee9a	+<ab1>	rdf:nil == rdf:nil
:c	+<c>	c has datatype xs:string

▶ Detalles

Schema Engine (current: ShEx) ShEx Schema embedded: □

RDF Data

```
@prefix :       <http://www.example.com> .
@prefix rdf:    <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

:c       :namec  "c" .

:nondeterministic1  a  ( :a :b ) .

:a       :namea  "a" .

:nondeterministic2  a  ( :a :c ) .

:b       :nameb  "b" .

```

Data Format TURTLE

Schema

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX xs: <http://www.w3.org/2001/XMLSchema#>
PREFIX : <http://www.example.com>

<nondeterministic> {
  a @<ab> OR @<ac> ;
}

<ab> {
  rdf:first @<a> ;
  rdf:rest @<ab1> ;
}

<ac> {
  rdf:first @<a> ;
  rdf:rest @<ac1> ;
}

<ab1> {
  rdf:first @<b> ;
  rdf:rest [rdf:nil]  ;
}

<ac1> {
  rdf:first @<c> ;
  rdf:rest [rdf:nil]  ;
}

<a> {
  :namea xs:string ;
}

<b> {
```

Schema Format ShExC

Inference before

Mode NONE

Trigger mode

Mode ShapeMap

Shape map

```
:nondeterministic1@<nondeterministic>,:nondeterministic2@<nondeterministic>
```

Shape map format COMPACT

Other options

Editor theme: Eclipse

Figure 3.3: Validation result using Shaclex validator of a ShEx schema converted from a non-deterministic XML Schema document. In the Shape map input area text we have indicated to Shaclex validator to check if :nondeterministic1 and :nondeterministic2 hold the form of shape ¡nondeterministic¿. In the top of the page the satisfactory result is shown in green.

```
<b> {
  :nameb xs:string ;
}

<c> {
  :namec xs:string ;
}
```

Listing 3.26: Non-Deterministic schema and its ShEx counterpart

These sequences are translated as shown in Section 3.4 and the final result can be seen in Listing 3.26. The question is that if this non-determinism is also transferred to the converted schemata. In order to check the actual behaviour we have run this example on Shaclex validator which shows that the validation is performed correctly (see Figure 3.3).

This behaviour is motivated by two things: firstly, the structure of RDF lists is different from XML Schema sequences which makes the validation to be performed in a different form; consequently, the validation in ShEx is performed recursively trying to match shape by shape. Therefore, if an element match with a shape this will scale up into the recursion tree without creating ambiguity problems.

3.7 Conclusions and Future work

In this work, a possible set of mappings between XML Schema and ShEx has been presented. With this set of mappings, automation of XML Schema conversions to ShEx is a new possibility for schema translation which is demonstrated by the prototype that has been developed and presented in this paper. Using an existing validator helped to demonstrate that an XML and its corresponding XML Schema are still valid when they are converted to RDF and ShEx.

One future line of work that should be tackled is the loss of semantics: with this kind of transformations some of the elements could not be converted back to their original XML Schema constructs. Nevertheless, it is a difficult problem due to the difference between ShEx and XML data models and it would involve some sort of modifications and additions to the ShEx semantics (like the previously mentioned inheritance).

To cover more business cases and make this solution more compatible with existing systems, there is the need to create mappings for Schematron and Relax NG as a future work. Relax NG is grammar-based but Schematron is rule based, which will make conversion from Relax NG to ShEx more straightforward than from Schematron to ShEx, as ShEx is also grammar-based. Another line of future work is to adapt the presented mappings to SHACL: most of the mappings follow a similar structure. Moreover, the rule-based Schematron conversion seems more feasible using the advanced SHACL-SPARQL features which allow to expand the core SHACL language by using SPARQL queries to validate complex constraints.

With the present work, validation of existing transformations between XML and RDF is now possible and convenient. This kind of validations makes the transformed data more reliable and trustworthy and it also facilitates migrations from non-semantic data formats to semantic data formats.

Conversions from other formats (such as JSON Schema, DDL, CSV Schema, etc.) will also be investigated to permit an improvement of data interoperability by reducing the technological gap.

Acknowledgments

This work has been partially funded by the Vice-rectorate for Research of the University of Oviedo under the call of "*Programa de Apoyo y Promoción de la Investigación 2017*".

Chapter 4

Enhancing e-Learning content by using Semantic Web technologies

This article was originally published as:

Herminio García-González, José Emilio Labra Gayo, and María del Puerto Paule Ruiz. Enhancing e-Learning Content by Using Semantic Web Technologies. *IEEE Trans. Learn. Technol.*, 10(4):544–550, 2017

Herminio García-González, José Emilio Labra Gayo and María del Puerto Paule Ruiz were with the Department of Computer Science, University of Oviedo, Oviedo, Asturias, Spain

The journal has the following metrics according to 2017 JCR:

- 2017 Impact Factor: 1.869
- 5 Year Impact Factor: 2.5
- Computer Science, Interdisciplinary Applications: Q2 (48/109)

Abstract

We describe a new educational tool that rely on Semantic Web technologies to enhance lessons content. We conducted an experiment with 32 students whose results demonstrate better performance when exposed to our tool in comparison with a plain native tool. Consequently, this prototype opens new possibilities in lessons content enhancement.

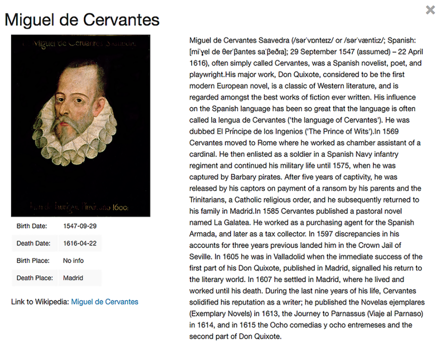
4.1 Introduction

E-Learning has supposed a huge advance in learning environments allowing educational community to rely on new technologies to give an improved experience and empower their students with better materials [99]. In this new era of learning, new learning environments have arisen such as Learning Management Systems (LMS) which enable users to share contents, create courses, collaborate with each other through forums or wikis, create and fulfil assignments, give and receive feedback and some others. They have been integrated in many universities as part of courses and degrees and many students and teachers are, nowadays, familiar with them. Nevertheless, with these novel tools new challenges arise. Among the diverse changes that may be covered in this area, we will focus on Semantic Web and content enhancement. Teachers contents on e-Learning platforms are contributing to enhance the knowledge of the attendants. But related with this main content there is more information that can be emerged using the appropriate tools. For example, if some content is mentioning Obama, a student may be wondering who is Obama or confused if Obama is mentioned in various ways (e.g. Barack, Obama, Barack Obama, B. Obama or even Barack Hussein Obama II). This problem is derived from the lack of semantics in the uploaded content. Our proposal is to take advantage of Semantic Web in order to: provide more information about outstanding entities, reconcile entities and enrich pages with RDFa[1] (Resource Description Framework in Attributes) and microformats. The main contribution of this work is a new technology that uses a set of Semantic Web techniques to complement and expand the learning courses content. This technology allows to enhance learning content hosted at LMS, favouring the increment of courses didactic effectiveness [100] as this work states.

4.2 Related work

The most similar architecture to that shown in our work is presented by [54], where authors show an architecture to enhance government data and then publish these enhanced data as Linked Data. An enhancement centred on museums was reported by [25], where authors use Semantic Web to link and add contents to museum objects. In [33], the authors propose an enhancement of user-generated content using geospatial Linked Open Data to improve tagging of Social Media platforms, like Facebook. The use of ontologies to recommend new personalised contents to the students depending on their fails and progress, is described in [55]. Enhancement for media management systems including videos, images and articles is described in [71] where they used a Red

[1] https://rdfa.info/



Miguel de Cervantes

Birth Date:	1547-09-29
Death Date:	1616-04-22
Birth Place:	No info
Death Place:	Madrid

Link to Wikipedia: Miguel de Cervantes

Miguel de Cervantes Saavedra (/sərˈvɒnteɪz/ or /sərˈvæntiːz/; Spanish: [miˈɣel de θerˈβantes saˈβeðra]; 29 September 1547 (assumed) – 22 April 1616), often simply called Cervantes, was a Spanish novelist, poet, and playwright.His major work, Don Quixote, considered to be the first modern European novel, is a classic of Western literature, and is regarded amongst the best works of fiction ever written. His influence on the Spanish language has been so great that the language is often called la lengua de Cervantes ('the language of Cervantes'). He was dubbed El Príncipe de los Ingenios ('The Prince of Wits').In 1569 Cervantes moved to Rome where he worked as chamber assistant of a cardinal. He then enlisted as a soldier in a Spanish Navy infantry regiment and continued his military life until 1575, when he was captured by Barbary pirates. After five years of captivity, he was released by his captors on payment of a ransom by his parents and the Trinitarians, a Catholic religious order, and he subsequently returned to his family in Madrid.In 1585 Cervantes published a pastoral novel named La Galatea. He worked as a purchasing agent for the Spanish Armada, and later as a tax collector. In 1597 discrepancies in his accounts for three years previous landed him in the Crown Jail of Seville. In 1605 he was in Valladolid when the immediate success of the first part of his Don Quixote, published in Madrid, signalled his return to the literary world. In 1607 he settled in Madrid, where he lived and worked until his death. During the last nine years of his life, Cervantes solidified his reputation as a writer; he published the Novelas ejemplares (Exemplary Novels) in 1613, the Journey to Parnassus (Viaje al Parnaso) in 1614, and in 1615 the Ocho comedias y ocho entremeses and the second part of Don Quixote.

Figure 4.1: Example of Miguel de Cervantes' card

Bull Content Pool for the demonstration. Using Semantic Web for interactive Relationship discovery is addressed in [78] where authors highlight its use in technology enhanced learning. In [81], the authors use Web Semantic mining techniques to provide different personalised e-Learning experiences. A use of Web Semantic to discover and share content in OpenCourseWare environments is described in [96]. Ontologies as a way for describing content, for defining learning material and for structuring learning material is presented in [120]. Annotating videos with Linked Open Data (LOD) vocabularies and therefore improve search of educational videos is described in [132].

Content enhancement has also been performed using adaptative techniques from the Adaptive Hypermedia proposed in [21] with differents approaches like the creation of adaptative languages [30] [93] [29] or using learning objects [32] [58].

4.3 Proposed prototype

We have developed a prototype called LODLearning to enhance lessons contents within LMS tools. Enhancements in this context refers to the addition and linking of related latent content into lessons material. That enhancement offers the opportunity to learn new knowledge without leaving the platform, providing the students with a new way of searching for related content. LODLearning performs a NLP (Natural Language Processing) entity recognition algorithm that extracts the most relevant known entities from the given text. It also searches through the Semantic Web for new content to add to these entities. Therefore, the principal idea behind LODLearning is to take advantage of the Semantic Web to complement and expand the learning content within courses.

Prototype use case

LODLearning takes the lessons content from the LMS tool and analyses it in order to retrieve meaningful entities that are shown to the user, enriching the present content with expanded information. For the hypothesis demonstration we have integrated the LODLearning prototype with Sakai LMS, which

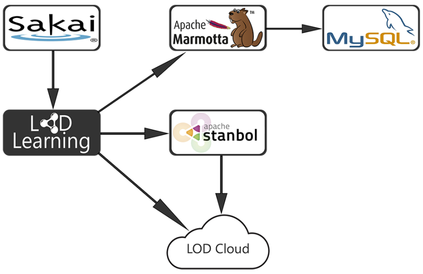
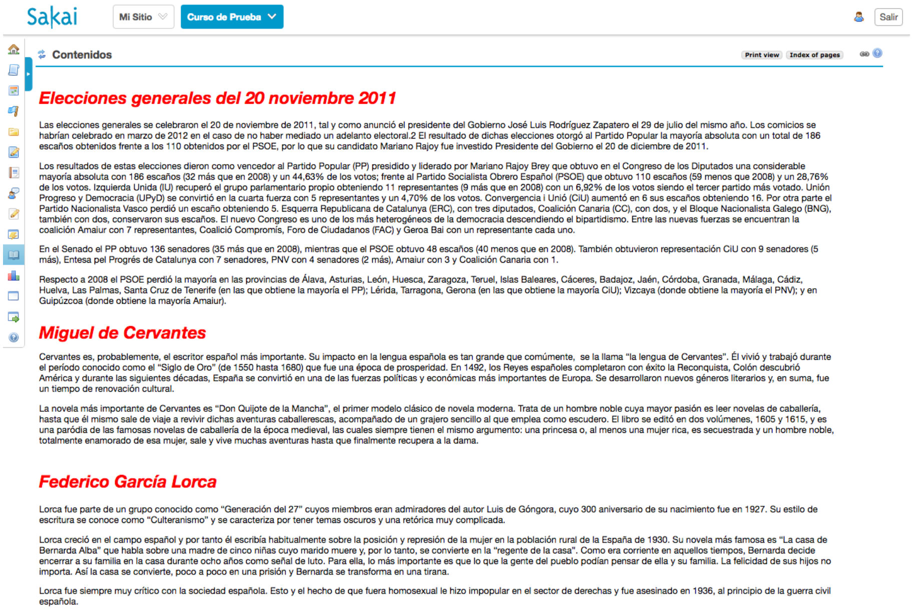


Figure 4.2: Component diagram of LODLearning prototype.



Elecciones generales del 20 noviembre 2011

Las elecciones generales se celebraron el 20 de noviembre de 2011, tal y como anunció el presidente del Gobierno José Luis Rodríguez Zapatero el 29 de julio del mismo año. Los comicios se habrían celebrado en marzo de 2012 en el caso de no haber mediado un adelanto electoral.2 El resultado de dichas elecciones otorgó al Partido Popular la mayoría absoluta con un total de 186 escaños obtenidos frente a los 110 obtenidos por el PSOE, por lo que su candidato Mariano Rajoy fue investido Presidente del Gobierno el 20 de diciembre de 2011.

Los resultados de estas elecciones dieron como vencedor al Partido Popular (PP) presidido y liderado por Mariano Rajoy Brey que obtuvo en el Congreso de los Diputados una considerable mayoría absoluta con 186 escaños (32 más que en 2008) y un 44,63% de los votos; frente al Partido Socialista Obrero Español (PSOE) que obtuvo 110 escaños (59 menos que 2008) y un 28,76% de los votos. Izquierda Unida (IU) recuperó el grupo parlamentario propio obteniendo 11 representantes (9 más que en 2008) con un 6,92% de los votos siendo el tercer partido más votado. Unión Progreso y Democracia (UPyD) se convirtió en la cuarta fuerza con 5 representantes y un 4,70% de los votos. Convergencia i Unió (CiU) aumentó en 6 sus escaños obteniendo 16. Por otra parte el Partido Nacionalista Vasco perdió un escaño obteniendo 5. Esquerra Republicana de Catalunya (ERC), con tres diputados, Coalición Canaria (CC), con dos, y el Bloque Nacionalista Galego (BNG), también con dos, conservaron sus escaños. El nuevo Congreso es uno de los más heterogéneos de la democracia descendiendo el bipartidismo. Entre las nuevas fuerzas se encuentran la coalición Amaiur con 7 representantes, Coalició Compromís, Foro de Ciudadanos (FAC) y Geroa Bai con un representante cada uno.

En el Senado el PP obtuvo 136 senadores (35 más que en 2008), mientras que el PSOE obtuvo 48 escaños (40 menos que en 2008). También obtuvieron representación CiU con 9 senadores (5 más), Entesa pel Progrés de Catalunya con 7 senadores, PNV con 4 senadores (2 más), Amaiur con 3 y Coalición Canaria con 1.

Respecto a 2008 el PSOE perdió la mayoría en las provincias de Álava, Asturias, León, Huesca, Zaragoza, Teruel, Islas Baleares, Cáceres, Badajoz, Jaén, Córdoba, Granada, Málaga, Cádiz, Huelva, Las Palmas, Santa Cruz de Tenerife (en las que obtiene la mayoría el PP); Lérida, Tarragona, Gerona (en las que obtiene la mayoría CiU); Vizcaya (donde obtiene la mayoría el PNV); y en Guipúzcoa (donde obtiene la mayoría Amaiur).

Miguel de Cervantes

Cervantes es, probablemente, el escritor español más importante. Su impacto en la lengua española es tan grande que comúmente, se la llama "la lengua de Cervantes". Él vivió y trabajó durante el período conocido como el "Siglo de Oro" (de 1550 hasta 1680) que fue una época de prosperidad. En 1492, los Reyes españoles completaron con éxito la Reconquista, Colón descubrió América y durante las siguientes décadas, España se convirtió en una de las fuerzas políticas y económicas más importantes de Europa. Se desarrollaron nuevos géneros literarios y, en suma, fue un tiempo de renovación cultural.

La novela más importante de Cervantes es "Don Quijote de la Mancha", el primer modelo clásico de novela moderna. Trata de un hombre noble cuya mayor pasión es leer novelas de caballería, hasta que él mismo sale de viaje a revivir dichas aventuras caballerescas, acompañado de un grajero sencillo al que emplea como escudero. El libro se editó en dos volúmenes, 1605 y 1615, y es una paródia de las famosas novelas de caballería de la época medieval, las cuales siempre tienen el mismo argumento: una princesa o, al menos una mujer rica, es secuestrada y un hombre noble, totalmente enamorado de esa mujer, sale y vive muchas aventuras hasta que finalmente recupera a la dama.

Federico García Lorca

Lorca fue parte de un grupo conocido como "Generación del 27" cuyos miembros eran admiradores del autor Luis de Góngora, cuyo 300 aniversario de su nacimiento fue en 1927. Su estilo de escritura se conoce como "Culteranismo" y se caracteriza por tener temas oscuros y una retórica muy complicada.

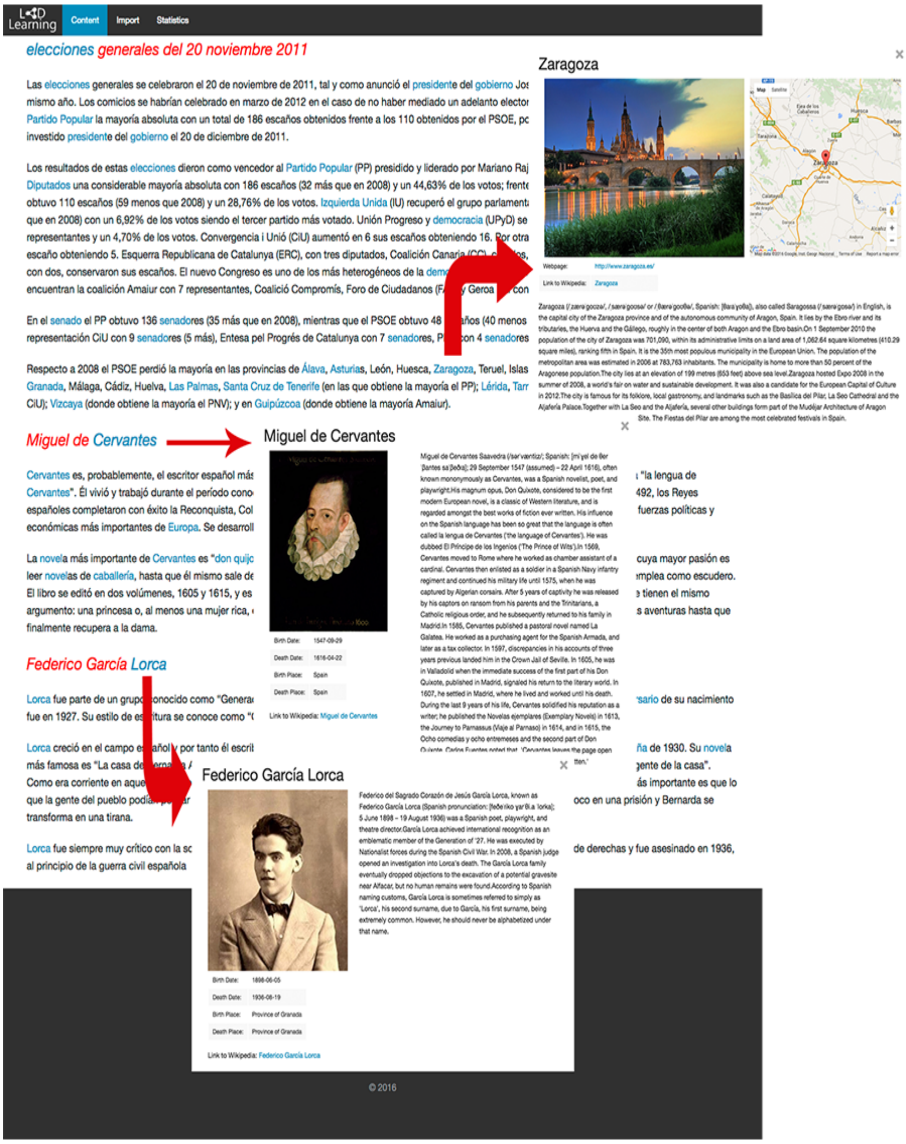
Lorca creció en el campo español y por tanto él escribía habitualmente sobre la posición y represión de la mujer en la población rural de la España de 1930. Su novela más famosa es "La casa de Bernarda Alba" que habla sobre una madre de cinco niñas cuyo marido muere y, por lo tanto, se convierte en la "regente de la casa". Como era corriente en aquellos tiempos, Bernarda decide encerrar a su familia en la casa durante ocho años como señal de luto. Para ella, lo más importante es que lo que la gente del pueblo podían pensar de ella y su familia. La felicidad de sus hijos no importa. Así la casa se convierte, poco a poco en una prisión y Bernarda se transforma en una tirana.

Lorca fue siempre muy crítico con la sociedad española. Esto y el hecho de que fuera homosexual le hizo impopular en el sector de derechas y fue asesinado en 1936, al principio de la guerra civil española.

Figure 4.3: Sakai Lessons tool with the three topics covered in the evaluation included.

supports all the learning management needed in a typical course environment providing different ways to integrate with and expand its functionality. In particular, it supports the IMS Learning Tools Interoperability protocol [114]). From a teacher point of view only a few steps are required for the enhancement of the content. The first time the teacher enters into the tool, lessons content will be displayed without any enhancement. If the teacher goes to the import section, an entity recognition algorithm will be executed, providing then the teacher with a list of checkable entities (i.e., if Cervantes is mentioned it will be displayed in the list). These entities are accompanied with a confidence percentage indicating the probability that they would indeed be present in the lessons content.

Once relevant entities have been selected by the teacher, they will be added



elecciones generales del 20 noviembre 2011

Las elecciones generales se celebraron el 20 de noviembre de 2011, tal y como anunció el presidente del gobierno Jos
mismo año. Los comicios se habrían celebrado en marzo de 2012 en el caso de no haber mediado un adelanto elector
Partido Popular la mayoría absoluta con un total de 186 escaños obtenidos frente a los 110 obtenidos por el PSOE, pc
investido presidente del gobierno el 20 de diciembre de 2011.

Los resultados de estas elecciones dieron como vencedor al Partido Popular (PP) presidido y liderado por Mariano Raj
Diputados una considerable mayoría absoluta con 186 escaños (32 más que en 2008) y un 44,63% de los votos; frente
obtuvo 110 escaños (59 menos que 2008) y un 28,76% de los votos. Izquierda Unida (IU) recuperó el grupo parlament
que en 2008) con un 6,92% de los votos siendo el tercer partido más votado. Unión Progreso y democracia (UPyD) se
representantes y un 4,70% de los votos. Convergencia i Unió (CiU) aumentó en 6 sus escaños obteniendo 16. Por otra
escaño obteniendo 5. Esquerra Republicana de Catalunya (ERC), con tres diputados, Coalición Canaria (CC)
con dos, conservaron sus escaños. El nuevo Congreso es uno de los más heterogéneos de la dem
encuentran la coalición Amaiur con 7 representantes, Coalició Compromís, Foro de Ciudadanos (F y Geroa con

En el senado el PP obtuvo 136 senadores (35 más que en 2008), mientras que el PSOE obtuvo 48 años (40 menos
representación CiU con 9 senadores (5 más), Entesa pel Progrés de Catalunya con 7 senadores, P con 4 senadores

Respecto a 2008 el PSOE perdió la mayoría en las provincias de Álava, Asturias, León, Huesca, Zaragoza, Teruel, Islas
Granada, Málaga, Cádiz, Huelva, Las Palmas, Santa Cruz de Tenerife (en las que obtiene la mayoría el PP); Lérida, Tarr
CiU); Vizcaya (donde obtiene la mayoría el PNV); y en Guipúzcoa (donde obtiene la mayoría Amaiur).

Zaragoza

Webpage:	http://www.zaragoza.es/
Link to Wikipedia:	Zaragoza

Zaragoza (/ˌzærəˈɡoʊzə/, /ˌsærəˈɡoʊsə/ or /θærəˈɡoʊθə/, Spanish: [θaɾaˈɣoθa]), also called Saragossa (/ˌsærəˈɡɒsə/) in English, is the capital city of the Zaragoza province and of the autonomous community of Aragon, Spain. It lies by the Ebro river and its tributaries, the Huerva and the Gállego, roughly in the center of both Aragon and the Ebro basin.On 1 September 2010 the population of the city of Zaragoza was 701,090, within its administrative limits on a land area of 1,062.64 square kilometres (410.29 square miles), ranking fifth in Spain. It is the 35th most populous municipality in the European Union. The population of the metropolitan area was estimated in 2006 at 783,763 inhabitants. The municipality is home to more than 50 percent of the Aragonese population.The city lies at an elevation of 199 metres (653 feet) above sea level.Zaragoza hosted Expo 2008 in the summer of 2008, a world's fair on water and sustainable development. It was also a candidate for the European Capital of Culture in 2012.The city is famous for its folklore, local gastronomy, and landmarks such as the Basílica del Pilar, La Seo Cathedral and the Aljafería Palace.Together with La Seo and the Aljafería, several other buildings form part of the Mudéjar Architecture of Aragon Site. The Fiestas del Pilar are among the most celebrated festivals in Spain.

Miguel de Cervantes

Cervantes es, probablemente, el escritor español más
Cervantes". Él vivió y trabajó durante el período cono
españoles completaron con éxito la Reconquista, Col
económicas más importantes de Europa. Se desarroll

La novela más importante de Cervantes es "don quijo
leer novelas de caballería, hasta que él mismo sale de
El libro se editó en dos volúmenes, 1605 y 1615, y es
argumento: una princesa o, al menos una mujer rica,
finalmente recupera a la dama.

"la lengua de
492, los Reyes
fuerzas políticas y

cuya mayor pasión es
emplea como escudero.
tienen el mismo
s aventuras hasta que

Miguel de Cervantes

Birth Date:	1547-09-29
Death Date:	1616-04-22
Birth Place:	Spain
Death Place:	Spain

Link to Wikipedia: Miguel de Cervantes

Miguel de Cervantes Saavedra (/sərˈvæntiːz/; Spanish: [miˈɣel de θerˈβantes saˈβeðra]; 29 September 1547 (assumed) – 22 April 1616), often known mononymously as Cervantes, was a Spanish novelist, poet, and playwright.His magnum opus, Don Quixote, considered to be the first modern European novel, is a classic of Western literature, and is regarded amongst the best works of fiction ever written. His influence on the Spanish language has been so great that the language is often called la lengua de Cervantes ('the language of Cervantes'). He was dubbed El Príncipe de los Ingenios ('The Prince of Wits').In 1569, Cervantes moved to Rome where he worked as chamber assistant of a cardinal. Cervantes then enlisted as a soldier in a Spanish Navy infantry regiment and continued his military life until 1575, when he was captured by Algerian corsairs. After 5 years of captivity he was released by his captors on ransom from his parents and the Trinitarians, a Catholic religious order, and he subsequently returned to his family in Madrid.In 1585, Cervantes published a pastoral novel named La Galatea. He worked as a purchasing agent for the Spanish Armada, and later as a tax collector. In 1597, discrepancies in his accounts of three years previous landed him in the Crown Jail of Seville. In 1605, he was in Valladolid when the immediate success of the first part of his Don Quixote, published in Madrid, signaled his return to the literary world. In 1607, he settled in Madrid, where he lived and worked until his death. During the last 9 years of his life, Cervantes solidified his reputation as a writer; he published the Novelas ejemplares (Exemplary Novels) in 1613, the Journey to Parnassus (Viaje al Parnaso) in 1614, and in 1615, the Ocho comedias y ocho entremeses and the second part of Don Quixote. Carlos Fuentes noted that, 'Cervantes leaves the page open tten.'

Federico García Lorca

Lorca fue parte de un grupo conocido como "Genera
fue en 1927. Su estilo de escritura se conoce como "

sario de su nacimiento

Lorca creció en el campo español y por tanto él escri
más famosa es "La casa de
Como era corriente en aque
que la gente del pueblo podía
transforma en una tirana.

ña de 1930. Su novela
gente de la casa".
ás importante es que lo
oco en una prisión y Bernarda se

Lorca fue siempre muy crítico con la sc
al principio de la guerra civil española

de derechas y fue asesinado en 1936,

Federico García Lorca

Federico del Sagrado Corazón de Jesús García Lorca, known as Federico García Lorca (Spanish pronunciation: [feðeˈɾiko ɣarˈθi.a ˈlorka]; 5 June 1898 – 19 August 1936) was a Spanish poet, playwright, and theatre director.García Lorca achieved international recognition as an emblematic member of the Generation of '27. He was executed by Nationalist forces during the Spanish Civil War. In 2008, a Spanish judge opened an investigation into Lorca's death. The García Lorca family eventually dropped objections to the excavation of a potential gravesite near Alfacar, but no human remains were found.According to Spanish naming customs, García Lorca is sometimes referred to simply as 'Lorca', his second surname, due to García, his first surname, being extremely common. However, he should never be alphabetized under that name.

Birth Date:	1898-06-05
Death Date:	1936-08-19
Birth Place:	Province of Granada
Death Place:	Province of Granada

Link to Wikipedia: Federico García Lorca

Figure 4.4: LODLearning tool with content enhancements. The arrows show the action performed when a link is pressed, revealing its corresponding enhanced content.

to the system that will finally show the lessons content with the enhancement added. This makes the enhancement system dynamic because it can be adapted depending on the content and the teacher requirements.

The system adds new enhanced content to lessons by using cards which show different information depending on the entity type previously selected by the teacher. New content to lessons can be added by using individual cards for every recognised entity. Cards can be designed with different information depending on entity type, for example, a photograph, a description, the birth date, the birth place, the death date, the death place and the wikipedia link for person entities. An example is shown in Figure 4.1.

For embedding these cards into the original content we opted for a modal based approach, showing a link when an entity is mentioned. When the link is pressed the corresponding item is displayed showing more information about the entity. With this approach new knowledge can be offered to the user without the need to leave the tool and the main content (see Figure 4.4).

Technological stack

The following technologies are being used:

- Sakai[2]: This is the LMS tool that is responsible for all the learning infrastructure. It offers authentication, course management, content management and an interface to expand its functionality.

- Apache Stanbol[3]: This component runs NLP and returns a list of URIs with some relevant attributes. Stanbol is used as an entity recogniser and entity disambiguator.

- Apache Marmotta[4]: This is a RDF (Resource Description Framework) triple store which offers a SPARQL[5] (SPARQL Protocol and RDF Query Language) endpoint and a set of web services for updating RDF content which is used to persist the enhanced content. Marmotta adapts a MySQL database to persist triples on it.

- DBpedia[6]: This project collects data from Wikipedia and transforms it into RDF. DBpedia is part of the LOD Cloud[7].

Figure 4.2 provides a diagram on how these technologies interact in our prototype. For the connection between Sakai and the prototype we used the LTI protocol [114], from the IMS Global Learning Consortium, in its 1.1 version. This protocol is a standard that defines how educational applications should communicate with LMSs. Between Apache Marmotta and LODLearning we used a REST API as well as between Apache Stanbol and LODLearning. DBpedia exposes a SPARQL endpoint which is queried with Apache Jena[8]. And finally, Apache Marmotta communicates with MySQL through JDBC.

[2] https://sakaiproject.org/

[3] https://stanbol.apache.org/

[4] http://marmotta.apache.org/

[5] https://www.w3.org/TR/rdf-sparql-query/

[6] http://wiki.dbpedia.org/

[7] http://lod-cloud.net/

[8] https://jena.apache.org

The application flow is the following: once the application is invoked from Sakai, it queries the Sakai lessons API and adapts the available menus depending on the user role (i.e., admin, instructor or student). If a teacher performs an entity content importation, LODLearning sends the lesson content to Apache Stanbol which executes an entity recognition algorithm. Once Stanbol finishes, it returns a RDF graph with the entities URI, the confidence and some extra attributes. LODLearning persists this RDF and some extra attributes (queries from the DBpedia) to Apache Marmotta. Finally, once the importation is persited, whenever a user enters to content section, LODLearning will run SPARQL queries for the different persisted entities. LODLearning will also change entities appearances for links that will reveal their cards.

4.4 Prototype evaluation

This evaluation is focused on the didactic effectiveness measurement of the enhanced content performed using Semantic Web technologies. In our study, didactic effectiveness is associated with the change in students' performance while they were using the tool [79].

For the evaluation we composed a lesson into the Sakai learning system which was formed by three different topics (Spanish General Elections of 2011, Miguel de Cervantes and Federico García Lorca). The Native Sakai tool with these topics can be seen in Figure 4.3. In contrast, LODLearning downloads these lessons and enhances them with related content about the current lesson which is shown in form of cards. These cards will later appear whenever a student performs a click in the corresponding link (see Figure 4.4).

Therefore, the main difference between Sakai native tool and LODLearning lies in that more optional content that can be consulted by the students and in the experience that the students get from both tools.

The sample comprised 32 students pursuing the mandatory education stage in a State High School from the North of Spain and consisted of 18 women and 14 men aged from 13 to 14 years. The sample was divided into two groups in a random manner, namely control and experimental groups to perform an inter-subject study.

Control group evaluation was carried out by means of two different tasks for an intra-subject study. The first one (pretest) consisted in a questionnaire about three different topics covered in the Sakai course lesson. This first questionnaire was completed without any tool exposition in order to assess the knowledge of the sample. Then, the control group was exposed to the Sakai lessons native tool where the students read and memorised the exposed contents to perform a second questionnaire (posttest) about these topics. The experimental group evaluation was performed with the same method. However, it was exposed to our own designed prototype. Finally, the sample was asked to complete a satisfaction questionnaire to know their impressions about the tools they were exposed. This procedure can be seen in Figure 4.5. Time intervals, for both groups, for the completion of every requested task were as follows: 10 minutes for the first and second questionnaires, 5 minutes for satisfaction questionnaire and 15 minutes for reading and memorising the exposed tool contents.

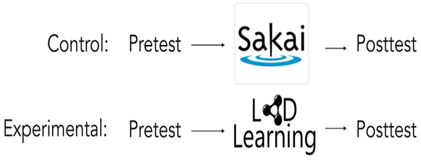


Figure 4.5: Evaluation process permorfed by the students in the evualuation of both tools.

Table 4.1: Marks obtained by the students. Sample size(n), mean($\bar{x}$), standard deviation(s), max and min for every group. 'Before' refers to results before exposition to the tool and 'After' to results after exposition to the tool.

	Control Before	Experimental Before	Control After	Experimental After
n	16	16	16	16
$\bar{x}$	14.77273	19.88636	32.89474	42.10526
s	9.889193	14.16889	13.79175	10.52632
max	36.360	54.550	57.89	57.89
min	0	0	10.53	21.05

The two evaluation questionnaires contained 11 and 19 questions respectively, all of them assigned with the value 1 for the right answer and 0 for no response or a wrong answer. Both of them displayed queries about present content in the lessons Sakai tool, either in the native version or in the content enhanced one using the LODLearning prototype. Questions were single choice or free text where some of the questions asked about multimedia content like maps and images. The first questionnaire consisted of 6 standard questions and 5 questions about the enhanced content. The second one included the first questionnaire plus 6 questions about the enhanced content and 2 standard questions. Satisfaction questionnaires —based on a Likert scale— were composed by 6 questions about the two different tools. For both groups the questionnaires were composed of the same questions. This satisfaction questionnaire was completed by 30 students out of the 32 ones. These 2 students preferred not to complete the satisfaction questionnaire. Questionnaires were designed and completed using the Google Docs platform and then downloaded as a CSV file for transformation and calculation of final marks with our own Python script. The technological stack described in the previous section was hosted in an Ubuntu 14.04 LTS server where students had access to it through internet by using Chrome or Firefox in their latest versions.

Results were collected adding 1 point for every correct student answer and then their marks were normalised in a 100 base following the English grading system. Results are shown in Table 4.1.

4.5 Results

Statistical analysis was performed using R, version 3.2.4 [107]. A Student's t-test was carried out between control and experimental groups (inter-subject

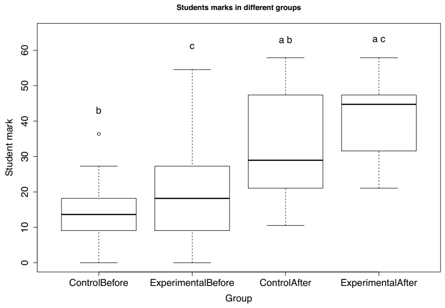


Figure 4.6: Representation of the experiment results for control and experimental groups after and before exposition to the tools. *b* & *c* very significant differences ($p < .001$). *a* significant differences ($p < .05$) by means of Student's t-test.

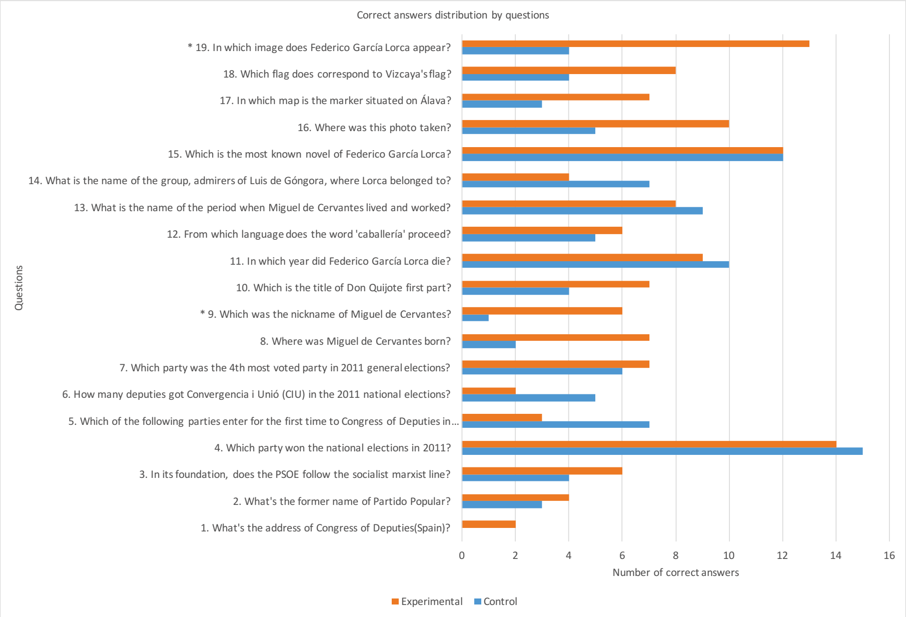


Figure 4.7: Distribution of correct answers by each question for control and experimental groups after exposition to the tool. Each bar represents the number of students that gave a correct answer for the respective question. * Significant evidence for Experimental > Control ($p < .05$) by means of Fisher's exact test.

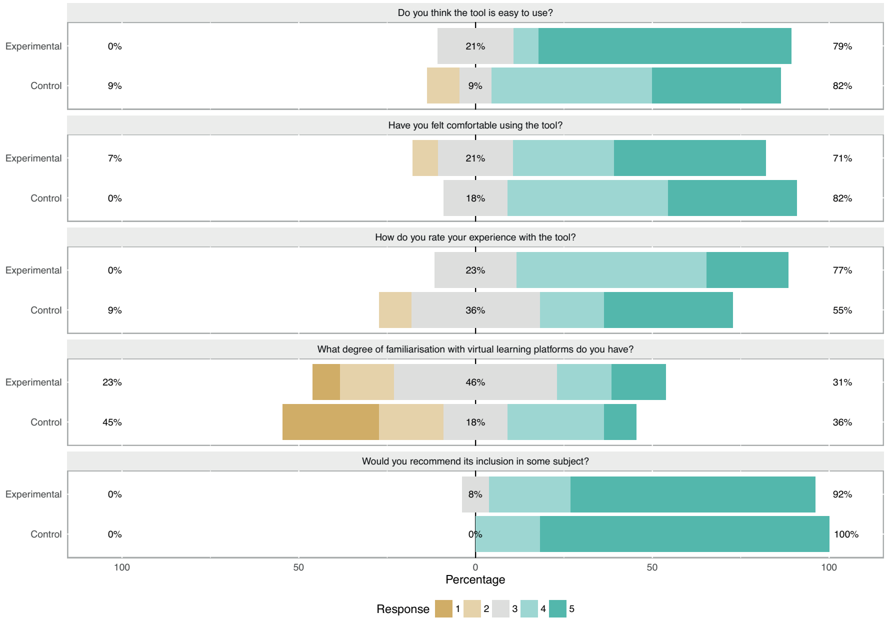


Figure 4.8: Control and experimental groups satisfaction punctuations about the two different tools in a Likert scale based questionnaire. Punctuation of 1 refers to Strongly disagree/Very poor and 5 to Strongly agree/Very good

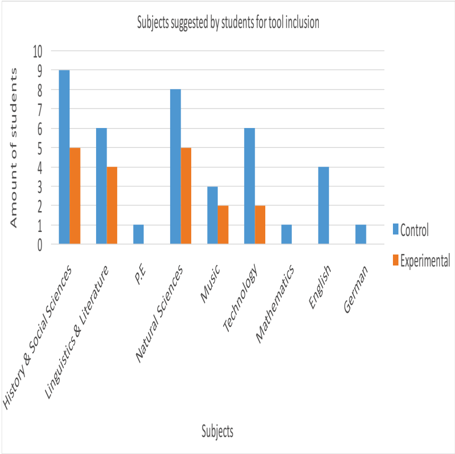


Figure 4.9: Bar chart which represents the number of students that suggested the inclusion of their tested tool in different subjects they were coursing.

study) before and after being exposed to the lessons based on Sakai native tool or to the LODLearning prototype, respectively; as well as between after and before within the same group (intra-subject study). Before exposition to the tool differences between control and experimental groups were not significant ($p = .24$). However, with the conventional level of significance ($\alpha = .05$), after exposition to the tool differences between control and experimental groups were significant ($p = .04$). Differences between the same group before and after exposition to the tool were very significant ($p = .00018$) and ($p = .00002$) for control group and experimental group respectively (see Figure 4.6). As part of the didactic effectiveness study for LODLearning, Cohen's d index [26] was also calculated to know the effect size between control and experimental groups after exposition to the tool ($d = .75$). A study for correct answers ratio for each question (after exposition to the tool) was performed by means of Fisher's exact test. Results, shown in Figure 4.7, exhibit that experimental group was significantly greater than control group for questions 9 ($p = .04484$) and 19 ($p = .004069$). Results of satisfaction test are shown in Figure 4.8 as well as students subject suggestions for tools inclusion, Figure 4.9.

4.6 Discussion and interpretation of results

Pretest study indicates that students in two groups had similar performance before exposition to the tools indicating similar levels of knowledge in both groups ($p = .24$). Nevertheless, posttest results report significative differences in control and experimental groups after exposition to the tools, pointing to changes in students' performance when using our prototype in comparison to the native Sakai tool. Moreover, the effect size (Cohen's d) shows that our results are not only significant, but are relevant and close to a big effect size. This measure proves that our prototype could be worthy to be used by its positive impact on students' performance. Another facet that deserves to be highlighted is the novelty aspect which can be a motivating factor and would stimulate students' interest. Scientific literature reports it in areas such as mathematics [67] and sciences [80]. However, this novelty aspect is present in both tools as students reflected in the degree of familiarisation question (see Figure 4.8).

When questions are considered separately some interesting data arise. Questions 19 and 9 suggest significant differences between control and experimental groups. Both of them were part of the enhanced data included into the prototype. Questions 19, 18, 17 and 16 registered the biggest differences; these questions, about multimedia items (i.e., maps and photographs), show that students tend to perform better with multimedia learning content. The other question with a significant difference between groups, question 9, suggests that when the prototype uses a short description text (e.g., question 9 and 8) students tend to remember this text more than when using a long text (e.g., question 2). Other questions about enhanced content (i.e., questions 12, 3, 2, 1) registered some better performance in the experimental group without as big differences as the previous ones which are caused also by long description text. However, questions 14, 13, 11, 6 and 5, about standard content, registered a better performance in the control group which might be influenced by the bigger amount of contents that should be memorised by the experimental group.

These results report that to obtain better content didactic effectiveness short text and multimedia content are the ones that should be prioritised which are in line with similar results reported by [100].

As Figure 4.8 shows, there are not significant differences among students in satisfaction levels. This supports that students are equally satisfied when using both tools. Therefore, we consider that LODLearning could be included in State High School courses without affecting notably students workflow with virtual learning environments. Moreover, these answers indicate no relevant issues in using the LODLearning prototype by the students. In contrast, LODLearning does not seem to increase satisfaction levels for the students nor their inclusion recommendation levels even though it does increase students' performance. However, this might be influenced by their degree of familiarisation with virtual learning platforms as they exposed in the first questions of the satisfaction questionnaire. These results also report that the enhancement content can be added transparently without interfering with the student and its learning process. Moreover, they rated their experience with the tools very positively and they also recommended their inclusion in subjects they were coursing.

When asked about their recommended subjects for tool inclusion they tended to recommend subjects related to the contents of this evaluation (i.e., History & Social Sciences and Linguistics & Literature) but also subjects like Natural Sciences and Technology where enhanced content about some difficult terms might be useful. These results are in line with the control suggestions where students tended to recommend more subjects, but most rated subjects are those which the experimental group recommended. The absence of Fine Arts draws attention, as it might be an interesting subject where it would be possible to conduct a more in-depth study.

One of the main lacks of other approaches is that the teacher needs to have technical knowledge [30] [93] [29] [32] [58]. However, in our approach, the teacher only needs to choose between the recognised entities in order to enhance the content. With our tool, and the support of Semantic Web, our approach provides more flexibility due to its design. Furthermore, other approaches did not cover a numerical evaluation [54] [25] [33] [71] [81] [120] nor a didactic effectiveness evaluation [55] [78] [96] [132], whereas our work includes this type of evaluation.

4.7 Conclusions and Future Work

In this work we have described the interaction of the LODLearning tool that we have developed and the way that it leverages the Linked Open Data Cloud to enhance lessons contents in the Sakai LMS. This prototype demonstrates that content enhancement can be used to improve courses didactic effectiveness. Nevertheless, support for more e-Learning platforms, inclusion of more enhancement content, an authoring system for designing new cards and more exhaustive and extended experiments should be addressed as future work in order to produce a better and more reliable platform. This work leads to a new way of use of Semantic Web Data in e-Learning platforms and highlights the combined use of e-Learning and Semantic Web in order to create more powerful learning tools.

Acknowledgments

This work has been funded by a collaboration with the Izertis company funded by the *Ministerio de Industria, Energía y Turismo* in the call of *Acción Estratégica de Economía y Sociedad Digital del 2014*. (TSI-100600-2014-44) Title: *Linked Open Data Learning (LOP): Enriqueciendo la experiencia formativa en eLearning*; the Department of Science and Innovation (Spain) under the National Program for Research, Development and Innovation: project EDU2014-57571-P; the European Union, through the European Regional Development Funds (ERDF); the Principality of Asturias, through its Science, Technology and Innovation Plan (grant GRUPIN14-100).

We also want to thank the State High School teachers: Modesto and María José, as well as their students for their willingness and collaboration with the experiment described in this work.

Chapter 5

Converting Asturian Notaries Public deeds to Linked Data using TEI and ShExML

This article was originally published as:

Herminio García-González, Elena Albarrán-Fernández, José Emilio Labra Gayo, and Miguel Calleja-Puerta. Converting Asturian Notaries Public deeds to Linked Data Using TEI and ShExML. In Alessandro Adamou, Enrico Daga, and Albert Meroño-Peñuela, editors, *Proceedings of the Third Workshop on Humanities in the Semantic Web (WHiSe 2020) co-located with 15th Extended Semantic Web Conference (ESWC 2020), Heraklion, Greece, June 2, 2020 (online)*, volume 2695 of *CEUR Workshop Proceedings*, pages 41–46. CEUR-WS.org, 2020

Herminio García-González and José Emilio Labra Gayo were with the Department of Computer Science, University of Oviedo, Oviedo, Asturias, Spain

Elena Albarrán-Fernández and Miguel Calleja-Puerta were with the Department of History, University of Oviedo, Oviedo, Asturias, Spain

The WHISE workshop was co-located with the Extended Semantic Web Conference 2020 (ESWC 2020) which is a CORE A conference according to CORE 2020 index[1].

[1] `http://portal.core.edu.au/conf-ranks/?search=ESWC&by=all&source=CORE2020&sort=atitle&page=1`

Abstract

Comprehension of past events and its reconstruction is one of the tasks performed by historians. With the introduction of computer-aided methods the way in which historians perform their work has been transformed. One of these inclusions is the Semantic Web which can act as an alternative for publication, conciliation, standardisation and integration. Asturian notaries public contracts are a valuable material to understand the society of this epoch, specially in the Middle Ages where a renovation process in the institution was taking place. Therefore, in this work we explore the transformation of TEI-based XML transcriptions of notarial contracts to RDF by means of an heterogeneous data mapping tool which can improve the mechanism to publish Linked Data from existing transcriptions.

5.1 Introduction

Elucidating past events and bringing them to our days is one of the tasks performed by historians which search evidences to reconstruct historical discourses. As in many fields, the introduction of computer aided methodologies has opened a new dimension in historical research in which has been coined as Digital Humanities. One field that has had a good reception is the Semantic Web whose technologies have been envisaged as a mean of publication, conciliation, standardisation and integration in the Humanities field [88].

A particular main problem, in Digital Humanities, is the generation of Linked Data from historical material. Although many procedures were proposed, most of them involve creating *ad-hoc* solutions that could only handle with a certain model, and modifications would need much effort or a software expert. Another problem is how to deal with heterogeneous models without creating one solution per schema. In the Semantic Web community new tools that try to deal with data transformation and, also with heterogeneity—of formats and data models—have appeared [31]. Moreover, they try to offer tools that can be used by domain experts without the constant implication of a software expert. Therefore, this kind of tools can bring a new dimension on Linked Data generation from historical sources.

Among other historical resources, notarial contracts are particularly valuable for the study of western mediterranean societies in the Middle Ages [1]. This kind of documents and the notarial institution itself represented the conjunction of romanist legal traits—inherited from Ancient times—in a profoundly religious culture that was already showing evident signs of transformation. In Asturias, a small northern region of the Castilian Crown, the overall renovation process of the notarial institution in the mid XIII century transformed its own writing tradition.

In this paper we describe our work on transforming transcriptions of medieval Asturian notarial contracts encoded with Text Encoding Inititive (TEI), how this can be achieved with heterogeneous data mapping tools and what are the challenges that this methodology poses.

5.2 Related Work

In the last years several works have explored the idea of transforming XML-based historical artefacts to Linked Data. This is the case of [103] which explores a conversion from XML/TEI to RDF/XML on historical documents, [57] which discusses the use of XTriples[2] to model, link and visualise XML corpora as Linked Data and [36] which presents a transformation of TEI-XML annotated Latin medieval texts to RDF by means of XSLT.

Regarding the notarial deeds, the most similar work to the one presented in this paper is presented in [40] where the authors produce a dataset of Notarial Archives in Valleta extracting entities, keyphrases and relations from notarial deeds. However, the extraction of entities using artificial intelligence techniques instead of basing the creation on transcriptions suppose a difference between both works.

Although these works explore different ways of transforming data to RDF, none of them tackle the use of heterogeneous data mapping tools, which are capable to integrate—in the same script—data in various forms and formats. This proposal could lead to a faster transformation process due to the centralisation in one tool, the higher flexibility against model changes and addition of other sources of information—with heterogeneous formats like: CSV, HTML, JSON, etc.—and the improvement on learning time from the domain experts.

5.3 Historical background

In the XIII century, during the reign of king Alfonso X, a renewed doctrine influenced the elaboration of a legal frame fitted to the times and the particularities of the Castilian Crown.

This frame included a new legal corpus and the transformation of judicial and documentary practices. In this context, the traditional scribes—most of them coming from clerical institutions—were replaced by public notaries.

For Asturias, an ancient kingdom located in the north of Castile, the new policy established by king Alfonso X meant several changes, as it was the king's intention to modernise not only the administration of his realm, but to transform rural areas into a more dynamic urban ones.

Public notaries assumed their predecessors' role with a whole new meaning: first, written culture no longer belonged exclusively to the Church; second, they offered their services to everyone, no matter their economical solvency or social background; third, their profession was defined by the law; thus, they recorded every single legal action and contract in the daily life of the Asturian society.

As no notarial registers from this early period remain today, we are working with documents issued by these notaries. Most of them were preserved by Asturian monasteries and cathedral, as they frequently used notaries' services to record their economical activities. Nowadays, a great amount of these documents are still guarded by an ecclesiastical institution—the monastery of San Pelayo of Oviedo[3] is one of the richest private archives in the region—

[2] https://xtriples.lod.academy/index.html

[3] http://sanpelayomonasterio.org/

while many are also preserved at the Archivo Histórico Nacional[4]—the biggest public-state archive in Spain.

5.4 Methodology

Asturian notarial contracts from XIII and XIV centuries are held by several private and public-state archives which, in some cases, can hinder their access. Furthermore, in many cases digitised versions are not available. But even with digitised versions, it is still to be proven that accuracy of promising state-of-the-art Optical Character Recognition (OCR) techniques [46] can be transposed to regional variations (i.e., differences in typographies can pose a problem to these OCR techniques). Therefore, the work of an editor is essential to transcribe the writings of this era.

As a first step manuscripts are transcribed to TEI-XML using vocabulary features plus some additions which cover diplomatic elements [1]. This first phase holds the digitised content plus some structure information about the manuscript itself and meta-data[5]. However, entities such as places and names are neither represented unambiguously nor linked with other existing entities. Therefore, this step corresponds with the creation phase of historical information life cycle as proposed by [88].

This version of manuscripts transcription can be queried and published but it has the problem of entities reconciliation and integration with other datasets. As a way to solve these lacks, the translation of these TEI-XML transcriptions to Linked Data is explored. In this work we have decided to use ShExML as it offers a simple syntax and, as being developed by two of the authors, it can be tuned if it is necessary.

5.5 Transformation process

The transformation process begins with the creation of the transformation script in ShExML syntax[6].

To create the data model we have taken the schema.org vocabulary to define the general attributes. This vocabulary, in its pending branch[7], offers new types that are suitable for generics attributes of works like the one presented in this paper. Therefore, the archive component type is used to model the content and meta-data of each TEI-XML transcription.

Some of these attributes require to have another type in the object part. For these cases a shape link is made which is a mechanism to define a new shape with a new form that will be linked to the upper one. This is the case of the `schema:locationCreated` which is a `schema:Place` and has a name and a link to a Linked Data Cloud[8] entity. Here, we have linked the `schema:sameAs` attribute with their Wikidata counterpart entity. This process is made using

[4] http://www.culturaydeporte.gob.es/cultura/areas/archivos/mc/archivos/ahn/portada.html

[5] An example manuscript transcribed to XML can be seen on: `https://github.com/albarranelena/AsturianNotaries/blob/master/AAA\_7.xml`

[6] Script available on: `http://herminiogg.github.io/whiseIII-paper-2020/notariesShort.shexml`

[7] http://pending.schema.org

[8] `https://lod-cloud.net/`

the ShExML matchers feature[9] which allows to replace a string for another string of our choice. For instance, the town of Avilés can be linked with its Wikidata[10] entry. Therefore, linking shapes we are able to create links between generated triples and model schema.org types inside ShExML. With the iterator nesting we are able to cover the tree structure and, also, multiple children from one parent which must be considered as a one triple generation per child.

Once this script is generated we can use the ShExML engine[11] to convert an arbitrary number of files following this encoding model to their RDF counterpart. To check the conversion presented in this paper we also offer an online demo[12] where we can upload the generated script and select the "Convert to RDF" option to generate the RDF output[13].

5.6 Limitations and challenges

Although this conversion can cover a lot of what is described in the TEI-XML transcription there are some limitations—which are also in line with some limitations encountered in TEI vocabulary and related formal ontologies derivatives [23]. The first problem was shown in the Office shape where the people belonging to an office cannot be represented using schema.org. The most likely relation is the schema:employee; however, the relations in medieval times cannot be understood as being an employee of an organisation but as a guild. It is also a problem that there are not defined relations between the different roles inside a notarial office and there is not a procedure to create these roles.

This problem increases when a diplomatic study is raised. In this case, modelling aspects such as the legal action described in the contract, the tradition of the act or the number and role of the participants cannot be achieved with the current vocabulary. Although this limitation do not restrict the conversion to Linked Data[14] and, it can also be queried through SPARQL queries, it is true that it can limit future inferences and, moreover, it could limit integration with other graphs which is the final goal of Linked Data.

Other ontologies like FRBR [102], NIE-INE[15], RiC[77] and ROAR[16] can define similar concepts to schema.org with more specificity or flexibility. However, they tend to focus in general concepts and meta-data but not on the domain specific content. To the best of our knowledge, there is no domain specific ontology nor vocabulary which defines this topic, and the closest one is the CEI [129] vocabulary which still do not define all the concepts present in our corpora. Therefore, in order to increase transformation inference capability and standardisation, it arises that a new ontology definition for this topic should be tackled.

[9] `http://shexml.herminiogarcia.com/spec/\#matcher`

[10] https://www.wikidata.org/wiki/Q14649

[11] https://github.com/herminiogg/shexml

[12] http://shexml.herminiogarcia.com/editor

[13] Full output available on:
`http://herminiogg.github.io/whiseIII-paper-2020/notariesShort.ttl`

[14] Full ShExML script with diplomatic features: `http://herminiogg.github.io/whiseIII-paper-2020/notariesFull.shexml`
Full RDF result: `http://herminiogg.github.io/whiseIII-paper-2020/notariesFull.ttl`

[15] `https://github.com/nie-ine/Ontologies`

[16] `https://leonvanwissen.nl/vocab/roar/docs/`

Another problem is how to identify and disambiguate persons' names which will require a mechanism to identify and disambiguate them from other people with the same name and surname. It is also problematic that these people are not registered in any other repository as may be, for example, the Kings of Spain (e.g.: Wikidata, DBpedia, etc.). This would involve the addition of an entity disambiguation mechanism in ShExML plus the creation of specific algorithms for this case. This kind of knowledge extraction from the text would imply in a simpler and faster process for the transcriber that can focus more on the transcription process and less in the identification and categorisation of entities.

5.7 Conclusions

In this work we have explored the possibility to apply heterogeneous data mapping tools to a TEI-based XML transcription of notarial contracts in order to convert them to RDF. This transformation was carried out using ShExML, which aims to offer a simple syntax to define these kinds of transformations, and using the schema.org vocabulary to assure the integration of these corpora with other existing or future resources. The process has shown that schema.org and other existing vocabularies are not able to synthesise what is needed for diplomatic studies. Therefore, we envisage the creation of a diplomatic ontology as an approach to cover this topic. Moreover, we highlight the need for an identification and disambiguation mechanism for person entities to favour further analyses.

Funding

This work has been partially funded by the Principality of Asturias through the Severo Ochoa call (grants BP17-29 and BP16-51) and by the Ministry of Economy, Industry and Competitiveness under the calls of "Programa Estatal de I+D+i Orientada a los Retos de la Sociedad" (project TIN2017-88877-R) and "Proyectos de I+D de Generación de Conocimiento" (project PGC2018-093495-B-100).

Chapter 6

Discussion, challenges and future work

Looking to the results of Chapter 2 we can see that there is certain level of improvement on the usability of heterogeneous data mapping languages. Firstly, it is demonstrated that in the case of first-time users, with programming background, there is an improvement when using ShExML against other proposals. It is also true that in the light of the results all the proposals have some aspects to improve in order to offer a better usability (e.g., the error reporting system). However, as a low sample size was used ($n = 17$), it would be very valuable if other studies are carried out to corroborate or not these early findings. Moreover, it will be interesting to also cover other kind of users to see which are the advantages and disadvantages of these tools on other kind of samples. All these studies would offer a broader view on the usability topic on heterogeneous data mapping languages and which should be the direction in which these languages should evolve.

Another topic which is very related with this one, but that could be interesting to non-experts users, is the evaluation of Graphical User Interfaces (GUIs) that are intended as a Domain Specific Language (DSL) wrapper (e.g., RMLEditor). This type of tools offer the possibility to create the mappings without having to deal with the syntax details which can be very promising for non programming background or non-experts users. However, there are a few questions that are worthy to mention.

- Are non-experts users—in the case of not knowing the data model background and other relative aspects—capable to deal successfully with this kind of tools? It is true that with training everyone can perform almost every task. However, it seems that the goal of these graphical tools is to ease the creation of mappings without the need of a specific training. But, it is to be proven that this goal is achieved. Other possibility is that these graphical tools are beneficial for domain experts users which have some data in non-semantic formats and want to transform them into RDF. Or, finally, in the hands of computer experts which might prefer text-based version as they tend to offer more control, speed and flexibility. Anyway, it is a question that might be interesting to answer.
- Is it better to invest time on developing text-based approaches or in

graphical alternatives? It is clear that in the case of being useful for all kind of users these tools can embrace more users. But in the case that they are only useful for domain experts and people with programming background should we invest time on text-based approaches and training or in developing a GUI.

- Which percentage of people will use each version? It is possible that non-experts users migrate to text-based approaches when they are familiar with the mechanism and the theory behind them. Therefore, is it possible that this migration happens? If true, this GUI will serve as a training application. So, depending on the final percentage we can take decisions on this topic.

These questions are interesting when planning how to evolve ShExML and other data mapping languages. Is it worthy to create a graphical user interface or is it better to include new functionality and characteristics on ShExML? It is not an easy task to solve these questions but an interesting one which would give us some light and help on how to plan the future work of this topic.

Talking about future characteristics, we mentioned in Chapter 1 that the validation of the produced output is a desired practice as it offers normalised, clean and reliable data sources. In Chapter 3 we introduced our research on the transformation automation of XML Schema schemata to ShEx schemata as a way to demonstrate that not only data can be converted but also its schema and validation rules conferring the output with valuable characteristics. In this paper we shown a possible conversion and demonstrated that it is possible to convert the schema and validate the converted data with it. However, there is a loss of semantics due to the difference in data models which makes that reverse conversion cannot be achieved without losing information.

In addition, a relation can be established between this kind of conversions and heterogeneous data mapping tools as it will be interesting to include validation transformation in them. Therefore, two procedures seem interesting to be explored:

- Automatic conversion: Once we are able to convert a schema into its corresponding counterpart, and that we are able to identify how each element is translated. It arises the idea of reusing this knowledge—extracted from this transformation—to not only transform the schema but to transform also data. It seems possible that using the link between an element in the origin schema and the destiny schema the transformation process can be build using this pre-existing information. Specifically, in ShExML and using the XMLSchema2ShEx conversion it would involve to take the XML Schema information to create the XPath query which would return the desired values, recover the information about the iteration needs and finally the own link with its ShEx element counterpart. Then, having the ShEx schema built from XML Schema we will be able to use the generated shapes and embed into them the recovered expressions.

- Aided conversion: Other possibility and a more flexible one is to use this information to allow for a guided conversion where users can have some decision on the process. This would be of much help in the aforementioned GUI version of heterogeneous data mapping tools. This mechanism would also help to solve the absence of name-spaces that could occur in XML schemata and that is almost necessary in a Knowledge Graph.

In Chapter 4 we have explored the use of existing knowledge graphs to enhance learning processes inside LMS systems. Specifically, we have combined the use of Apache Stanbol as an entity disambiguation mechanism from which we extracted the most prominent entity IRIs from a given text. Then, querying the DBpedia knowledge graph we obtained additional knowledge that was added to the text content using cards. This technique shown an improvement on didactic effectiveness in contrast with the former method. This study opens a new possibility where the students can expand their knowledge on the topics that are more interesting for them but starting from the same point which is the provided text. As future work, more kind of entities can be supported and also, with the improvement of entity disambiguation mechanisms more accurate predictions will be possible.

This is a demonstration of what can be achieved using Semantic Web technologies and the power they have. Specifically, the mapping of heterogeneous data sources into new or existing knowledge graphs poses a possibility of expanding the LOD cloud to new levels. Then, this knowledge can be used into multiple fields—like the e-Learning case that we have shown—which can take benefit from it. Moreover, the inclusion and use of Semantic Web technologies have been seen also as a mean of publication, conciliation, standardisation and integration. Namely, these characteristics are very present in the Digital Humanities field [88]. This is also the possibility that we have explored in Chapter 5 where we applied the ShExML engine to a corpus of notaries public deeds—that were previously transcribed into XML-TEI—in order to transform the content to Linked Data. Using the schema.org vocabulary we were able to integrate the meta-data information about the manuscripts with other existing ones. However, the limitation of this vocabulary regarding the diplomatic features arises the need for a proper ontology able to give the required semantics for this field. Moreover, to empower future analysis made from this transformed corpus we envisage the use of entity disambiguation mechanisms which could solve the problematic of identifying persons and places. The inclusion in ShExML of mechanisms and tools explored in Chapter 4 could be one of the approaches to solve this problem. Moreover, this would be interesting not only for this particular case but for the vast majority of transformations which would want to integrate with existing knowledge graphs and vocabularies.

The process followed in Chapter 5 shows that ShExML and related tools can be very valuable tools when following FAIR principles as we mentioned in Chapter 1.

These two use cases show the importance that heterogeneous data mapping tools can have on many fields and, moreover, the importance that can have the whole Semantic Web in outer fields and how it can contribute to their advance. This whole work give us the idea that we must advance Semantic Web technologies but also we have to introduce them into other fields and see how we can contribute to them and learn from them. Then, we will be able to gain valuable lessons and improve our own processes and technologies. Needless to mention, this is also applicable to the whole computer science field.

Capítulu 7

Conclusiones

Nesta tesis esploremos el campu de l'integración de datos per aciu del diseñu y desarrollu de ShExML que tien por oxetivu facilitar esta xera pa los usuarios que principien nesti tipu llabores. En resultes del experimentu fechu nel marcu d'esti trabayu, podemos dicir que ShExML ameyora la realización d'estes llabores pa esti tipu d'usuarios en comparanza con otres ferramientes homólogues.

Amás, propusimos la conversión de formatos de validación —por exemplu, XMLSchema— a sos equivalentes na Web Semántica de manera que nun realicemos solo una tresformación de datos sinón tamién de los sos esquemes, ofreciendo un conxuntu d'atributos de calidá deseables nos datos, que puean aumentar la confianza n'ellos.

Les teunoloxies de la Web Semántica son candidates ideales pa ser aplicaes n'otros campos col fin d'ameyorar sos ferramientes y procesos. Nesta tesis investiguemos les meyores qu'estes teunoloxies puen ofrecer nel campu del E-Learning y les Humanidáes Dixitales. Nel primeru l'arriquecimientu de los testos formativos demostró la meyora de la efectividá didáutica de los recursos ufiertaos en comparanza cola ferramienta ofrecía pel software educativu. Nel campu de les humanidáes dixitales esploremos como ShExML pue ofrecer un mediu pa la tresformación de les trescripciones históriques en RDF, de manera qu'estes conversiones puean ayudar a la so normalización ya integración con otros materiales semeyantes.

Asina mesmu, s'indentificaron dellos puntos de meyora y propunximos diferentes retos y preguntes qu'esta llinea d'investigación debe responder nel futuru pa saber per onde debe siguir avanzando. D'ente ellos se pue destacar la decisión ente avanzar peles alternatives textuales o crear versiones gráfiques que puean algamar más tipos d'usuarios. Esti tipu de decisiones requieren de diversos estudios pa saber cual ye la meyor direición a la hora de siguir col desarrollu d'estes propuestes.

Col desarrollu d'estes ferramientes s'intenta minimizar el coste de la tresformación y migración de teunoloxíes non semántiques a teunoloxíes semántiques, permitiendo que muncha información puea ser integrada cola ñube de datos enllazaos. Asina, contribuyese a la integración del conocimientu, la so desambiguación y la so posible divulgación, siendo esto finalmente un fechu que pue redundar na sociedá.

D'esta tesis despréndese la importancia de l'investigación nesti tipu de soluciones y como esta temática pue ameyorar non solo los propios procesos de les ciencies de la computación sinón tamién d'otros campos que faen usu de les teunoloxíes creaes nésti. Porque, en resultancia, el futuru de les ciencies de la computación nun pasa pol crecimientu dixebráu sinón pela cooperación y arriquecimientu conxuntu con otros campos.

7.1 Conclusions (translation to English)

In this thesis we have explored data integration field by means of the design and development of ShExML which objective is to ease the data integration task to users which start with these kinds of tasks. In the light of the experiments results we can say that ShExML improved the realisation of these tasks by the users in comparison with other similar alternatives.

In addition, we have proposed the transformation of data validation formats (e.g., XMLSchema) to their equivalent formats in the Semantic Web. Therefore, we are not only performing a data transformation but also a transformation of their validation schemata which offers a set of desired quality attributes in the data. This process can improve users' confidence in them.

The Semantic Web technologies are great candidates to be applied in other fields with the purpose to ameliorate their tools and processes. In this thesis we have studied the improvements that these technologies can bring to the E-Learning and Digital Humanities fields. In the first one, the learning content enhancement has demonstrated the resources didactic effectiveness improvement in comparison with the tool offered by the educational software. In the Digital Humanities field we have explored how ShExML can offer a means for historical transcriptions transformation which can help to their normalisation and integration with other similar material.

Likewise, we have identified some points of improvement and we have proposed some challenges and questions that this research line should answer in the future to elucidate how it should evolve. We can highlight the decision to advance between the textual or graphical approaches which, the latter, can cover more type of users. This kind of decision requires different studies to know which is the better direction to take in the future.

With the development of these tools we try to minimise the transformation and migration costs from non-semantic alternatives to semantic ones, allowing that a lot of information can be integrated in the Linked Data Cloud. In this way, we can contribute to knowledge integration, its disambiguation, and its potential dissemination which may redounds in the society.

From this thesis we can deduce the importance of the research in these kinds of solutions and how this topic can enhance not only the computer science processes but also those that make use of the technologies created in it. Because, as a result, the future of the computer science field is not in an isolated growth but in the joint cooperation and enrichment with other fields.

Financiación

Esta tesis ha sido financiada principalmente por los siguientes organismos:

- Principado de Asturias: La Consejería de Educación y Cultura financió esta tesis dentro del marco del Programa de Ayudas ”Severo Ochoa”para la formación en investigación y docencia del Principado de Asturias en su convocatoria de 2017. Referencia: BP17-29.
- Universidad de Oviedo: El Vicerrectorado de Investigación a través de la convocatoria Programa de Apoyo y Promoción de la Investigación”bajo la Modalidad A: Ayudas puente para la consecución de ayudas de doctorado de carácter competitivo para el ejercicio 2017, en régimen de concurrencia competitiva.
- Ministerio de Economía, Industria y Competitividad: bajo la convocatoria del Programa Estatal de I+D+i Orientada a los Retos de la Sociedad. Referencia: TIN2017-88877-R.
- Ministerio de Industria, Energía y Turismo: en la convocatoria de Acción Estratégica de Economía y Sociedad Digital del 2014 bajo el proyecto de ”Linked Open Data Learning (LOP): Enriqueciendo la experiencia formativa en eLearning”. Referencia: TSI-100600-2014-44.
- CPER Nord-Pas de Calais/FEDER DATA bajo el título .^{A}dvanced data science and technologies 2015-2020”.
- ANR project Data Cert con referencia: ANR-15-CE39-0009.

Bibliography

[1] Elena Albarrán-Fernández. A TEI-Based model to encode notarial charters (Asturias, 1260-1350 ca .), September 2019.

[2] Kevin Ashton et al. That 'internet of things' thing. *RFID journal*, 22(7):97–114, 2009.

[3] Sören Auer, Sebastian Dietzold, Jens Lehmann, Sebastian Hellmann, and David Aumueller. Triplify: light-weight linked data publication from relational databases. In *Proceedings of the 18th International Conference on World Wide Web, WWW 2009, Madrid, Spain, April 20-24, 2009*, pages 621–630, 2009.

[4] Steve Battle. Round-tripping between XML and RDF. In *International Semantic Web Conference (ISWC), Hiroshima, Japan*, 2004.

[5] Steve Battle. Round-tripping between XML and RDF. In Jeremy J. Carroll, editor, *International Semantic Web Conference (ISWC)*, Posters, Hiroshima, November 2004.

[6] Steve Battle. Gloze: XML to RDF and back again. In *Proceedings of the First Jena User Conference*, 2006.

[7] Steve Battle. Gloze: XML to RDF and back again. In *Proceedings of the First Jena User Conference*, HP Labs, Bristol, May 2006.

[8] Tim Berners-Lee, James Hendler, Ora Lassila, et al. The semantic web. *Scientific american*, 284(5):28–37, 2001.

[9] Diego Berrueta, Jose Emilio Labra Gayo, and Ivan Herman. XSLT + SPARQL: Scripting the semantic web with SPARQL embedded into XSLT stylesheets. In Christian Bizer, Sören Auer, Gunnar Aastrand, and Grimnes Tom Heath, editors, *4th Workshop on Scripting for the Semantic Web*, volume 368, Tenerife, June 2008. CEUR-WS.

[10] David M. Berry. *Understanding digital humanities.* Springer, 2012.

[11] Geert Jan Bex, Frank Neven, and Jan den Bussche. DTDs versus XML schema: a practical study. In Luis Gravano and Sihem Amer-Yahia, editors, *Proceedings of the 7th international workshop on the web and databases: colocated with ACM SIGMOD/PODS 2004*, ICPS, pages 79–84, Paris, June 2004. ACM. doi: 10.1145/1017074.1017095.

[12] Paul V. Biron, Ashok Malhotra, World Wide Web Consortium, et al. XML Schema part 2: Datatypes. `https://www.w3.org/TR/xmlschema-2/`, 2004.

[13] Stefan Bischof, Stefan Decker, Thomas Krennwallner, Nuno Lopes, and Axel Polleres. Mapping between RDF and XML with XSPARQL. *Journal on Data Semantics*, 1(3):147–185, 2012.

[14] Stefan Bischof, Stefan Decker, Thomas Krennwallner, Nuno Lopes, and Axel Polleres. Mapping between RDF and XML with XSPARQL. *Journal on Data Semantics*, 1(3):147–185, 2012. doi: 10.1007/s13740-012-0008-7.

[15] Christian Bizer and Andy Seaborne. D2rq-treating non-rdf databases as virtual rdf graphs. In *Proceedings of the 3rd international semantic web conference (ISWC2004)*, volume 2004. Proceedings of ISWC2004, 2004.

[16] Hannes Bohring and Sören Auer. Mapping XML to OWL Ontologies. In Klaus P. Jantke, Klaus-Peter Fähnrich, and Wolfgang S. Wittig, editors, *Marktplatz Internet: von E-Learning bis E-Payment, 13. Leipziger Informatik-Tage*, volume 72 of *LNI*, pages 147–156, Leipzig, September 2005. GI.

[17] Iovka Boneva, Jose Emilio Labra Gayo, and Eric Prud'hommeaux. Semantics and Validation of Shapes Schemas for RDF. In Claudia d'Amato, Miriam Fernandez, Valentina Tamma, Freddy Lecue, Philippe Cudré-Mauroux, Juan Sequeda, Christoph Lange, and Jeff Heflin, editors, *International Semantic Web Conference*, volume 10587 of *Lecture Notes in Computer Science*, pages 104–120, Vienna, October 2017. Springer Verlag. doi: 10.1007/978-3-319-68288-4_7.

[18] Frank Breitling. A standard transformation from XML to RDF via XSLT. *Astronomische Nachrichten*, 330(7):755–760, 2009.

[19] Frank Breitling. A standard transformation from XML to RDF via XSLT. *Astronomische Nachrichten*, 330(7):755–760, 2009. doi: 10.1002/asna.200811233.

[20] Dan Brickley and R.V. Guha. RDF Schema 1.1. `https://www.w3.org/TR/rdf-schema/`, 2014.

[21] Peter Brusilovsky. Methods and techniques of adaptive hypermedia. *User modeling and user-adapted interaction*, 6(2-3):87–129, 1996.

[22] Peter Brusilovsky. *Adaptive Hypertext and Hypermedia*, chapter Methods and Techniques of Adaptive Hypermedia, pages 1–43. Springer Netherlands, Dordrecht, 1998.

[23] Fabio Ciotti, Silvio Peroni, Francesca Tomasi, and Fabio Vitali. An OWL 2 formal ontology for the text encoding initiative. In *Digital Humanities 2016, DH 2016, Conference Abstracts, Jagiellonian University & Pedagogical University, Krakow, Poland, July 11-16, 2016*, pages 151–153. Alliance of Digital Humanities Organizations (ADHO), 2016.

[24] James Clark and Makoto Murata. Relax NG specification. `http://relaxng.org/spec-20011203.html`, 2001.

[25] Mauro Coccoli and Ilaria Torre. Interacting with annotated objects in a Semantic Web of Things application. *Journal of Visual Languages & Computing*, 25(6):1012–1020, dec 2014.

[26] Jacob Cohen. Statistical power analysis for the behavioral sciences. Vol. 2. *Lawrence Earlbaum Associates, Hillsdale, NJ*, 1988.

[27] Richard Cyganiak, David Wood, and Markus Lanthaler. RDF 1.1 Concepts and Abstract Syntax. https://www.w3.org/TR/rdf11concepts/, feb 2014.

[28] Souripriya Das, Seema Sundara, and Richard Cyganiak. R2RML: RDB to RDF Mapping Language. `https://www.w3.org/TR/r2rml/`, 2012.

[29] Paul De Bra, Evgeny Knutov, David Smits, Natalia Stash, and Vinicius F C Ramos. GALE: A Generic Open Source Extensible Adaptation Engine. *New Rev. Hypermedia Multimedia*, 19(2):182–212, 2013.

[30] Paul De Bra, David Smits, Kees van der Sluijs, Alexandra I Cristea, Jonathan Foss, Christian Glahn, and Christina M Steiner. *Intelligent and Adaptive Educational-Learning Systems: Achievements and Trends*, chapter GRAPPLE: Learning Management Systems Meet Adaptive Learning Environments, pages 133–160. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[31] Ben De Meester, Pieter Heyvaert, Ruben Verborgh, and Anastasia Dimou. Mapping languages analysis of comparative characteristics. In *First Knowledge Graph Building Workshop, part of ESWC2019*, pages 1–8, 2019.

[32] M del Puerto Paule Ruiz, M Jesús Fernández Díaz, Francisco Ortín Soler, and Juan Ramón Pérez Pérez. Adaptation in current e-learning systems. *Computer Standards & Interfaces*, 30(1–2):62–70, 2008.

[33] Dong-Po Deng, Guan-Shuo Mai, Cheng-Hsin Hsu, Chin-Lung Chang, Tyng-Ruey Chuang, and Kwang-Tsao Shao. *Linking Open Data Resources for Semantic Enhancement of User–Generated Content*, volume 7774 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, jan 2013.

[34] Davy Van Deursen, Chris Poppe, Gäetan Martens, Erik Mannens, and Rik Van de Walle. XML to RDF Conversion: A Generic Approach. In *Automated solutions for Cross Media Content and Multi-channel Distribution, 2008. AXMEDIS '08. International Conference on*, pages 138–144, Washington, nov 2008.

[35] Davy Van Deursen, Chris Poppe, Gäetan Martens, Erik Mannens, and Rik Van de Walle. XML to RDF Conversion: A Generic Approach. In Paolo Nesi, Kia Ng, and Jaime Delgado, editors, *2008 International Conference on Automated solutions for Cross Media Content and Multichannel Distribution.*, pages 138–144, Florence, November 2008. IEEE. doi: 10.1109/AXMEDIS.2008.17.

[36] Molka Tounsi Dhouib, Catherine Faron Zucker, Arnaud Zucker, Olivier Corby, Catherine Jacquemard, Isabelle Draelants, and Pierre-Yves Buard. Transformation et visualisation de données rdf à partir d'un corpus annoté de textes médiévaux latins. In *IHM'14, 26e conférence francophone sur l'Interaction Homme-Machine*, Lille, France, Oct. 2014.

[37] Anastasia Dimou, Miel Vander Sande, Pieter Colpaert, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. RML: A generic language for integrated RDF mappings of heterogeneous data. In *Proceedings of the Workshop on Linked Data on the Web co-located with the 23rd International World Wide Web Conference (WWW 2014), Seoul, Korea, April 8, 2014.*, 2014.

[38] Nick Drummond, Alan L Rector, Robert Stevens, Georgina Moulton, Matthew Horridge, Hai Wang, and Julian Seidenberg. Putting OWL in order: Patterns for sequences in OWL. In Bernardo Cuenca Grau, Pascal Hitzler, Conor Shankey, and Evan Wallace, editors, *Proceedings of the OWLED'06 Workshop on OWL: Experiences and Directions*, Athens, Georgia, November 2006. CEUR-WS.

[39] TEI Consortium, eds. TEI P5: Guidelines for Electronic Text Encoding and Interchange. `http://www.tei-c.org/Guidelines/P5/`, 2017.

[40] Charlene Ellul, Joel Azzopardi, and Charlie Abela. Notarypedia: A knowledge graph of historical notarial manuscripts. In *On the Move to Meaningful Internet Systems: OTM 2019 Conferences*, pages 626–645, Cham, 2019. Springer International Publishing.

[41] Ivan Ermilov, Sören Auer, and Claus Stadler. CSV2RDF: User-driven CSV to RDF mass conversion framework. In *Proceedings of the ISEM*, volume 13, pages 04–06, Graz, Austria, 2013.

[42] Ronald Fagin, Phokion G. Kolaitis, Renée J. Miller, and Lucian Popa. Data exchange: semantics and query answering. *Theor. Comput. Sci.*, 336(1):89–124, 2005.

[43] Matthias Ferdinand, Christian Zirpins, and David Trastour. Lifting XML Schema to OWL. In Nora Koch, Piero Fraternali, and Martin Wirsing, editors, *Web Engineering: 4th International Conference, ICWE 2004*, volume 3140 of *Lecture Notes in Computer Science*, pages 354–358, Munich, July 2004. Springer Berlin Heidelberg. doi: 10.1007/978-3-540-27834-4_44.

[44] Andy Field. *Discovering statistics using IBM SPSS statistics.* Sage, 2013.

[45] Manuel Fiorelli, Tiziano Lorenzetti, Maria Teresa Pazienza, Armando Stellato, and Andrea Turbati. Sheet2rdf: a flexible and dynamic spreadsheet import&lifting framework for RDF. In *Current Approaches in Applied Artificial Intelligence - 28th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2015, Seoul, South Korea, June 10-12, 2015, Proceedings*, pages 131–140, 2015.

[46] Donatella Firmani, Paolo Merialdo, Elena Nieddu, and Simone Scardapane. In codice ratio: OCR of handwritten latin documents using deep convolutional networks. In *Proceedings of the 11th International Workshop on Artificial Intelligence for Cultural Heritage co-located with the 16th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2017), Bari, Italy, November 14, 2017*, pages 9–16, 2017.

[47] P. N. Fox, R. Mead, M. Talbot, and J. D. Corbett. Data management and validation. In *Statistical Methods for Plant Variety Evaluation*, pages 19–39, Dordrecht, 1997. Springer Netherlands.

[48] Fellipe Freire, Crishane Freire, and Damires Souza. Enhancing JSON to RDF data conversion with entity type recognition. In *Proceedings of the 13th International Conference on Web Information Systems and Technologies, WEBIST 2017, Porto, Portugal, April 25-27, 2017*, pages 97–106, 2017.

[49] Herminio García-González, Elena Albarrán-Fernández, José Emilio Labra Gayo, and Miguel Calleja-Puerta. Converting Asturian Notaries Public deeds to Linked Data Using TEI and ShExML. In Alessandro Adamou, Enrico Daga, and Albert Meroño-Peñuela, editors, *Proceedings of the Third Workshop on Humanities in the Semantic Web (WHiSe 2020) co-located with 15th Extended Semantic Web Conference (ESWC 2020), Heraklion, Greece, June 2, 2020 (online)*, volume 2695 of *CEUR Workshop Proceedings*, pages 41–46. CEUR-WS.org, 2020.

[50] Herminio García-González, Daniel Fernández-Álvarez, and José Emilio Labra Gayo. ShExML: An Heterogeneous Data Mapping Language based on ShEx. In *Proceedings of the EKAW 2018 Posters and Demonstrations Session co-located with 21st International Conference on Knowledge Engineering and Knowledge Management (EKAW 2018), Nancy, France, November 12-16, 2018.*, pages 9–12, 2018.

[51] Herminio García-González and José Emilio Labra Gayo. XMLSchema2ShEx: Converting XML validation to RDF validation. *Semantic Web*, 11(2):235–253, 2020.

[52] Herminio García-González, José Emilio Labra Gayo, and María del Puerto Paule Ruiz. Enhancing e-Learning Content by Using Semantic Web Technologies. *IEEE Trans. Learn. Technol.*, 10(4):544–550, 2017.

[53] Herminio García-González, Iovka Boneva, Sławek Staworko, José Emilio Labra-Gayo, and Juan Manuel Cueva Lovelle. ShExML: improving the usability of heterogeneous data mapping languages for first-time users. *PeerJ Computer Science*, 6(e318), 2020.

[54] Dietmar Glachs, Violeta Damjanovic, Felix Strohmeier, and Sergio Fernández. EAGLE–Local Government Learning Platform. *Proceedings of the Linked Learning meets LinkedUp: Learning and Education with the Web of Data, co-located with 13th International Semantic Web Conference (ISWC 2014).*

[55] Anatoly Gladun, Julia Rogushina, Francisco García-Sanchez, Rodrigo Martínez-Béjar, and Jesualdo Tomás Fernández-Breis. An application of intelligent techniques and semantic web technologies in e-learning environments. *Expert Systems with Applications*, 36(2):1922–1931, mar 2009.

[56] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview (Second Edition). `https://www.w3.org/TR/owl2-overview/`, 2012.

[57] Max Grüntgens and Torsten Schrade. Data repositories in the humanities and the semantic web: modelling, linking, visualising. In *1st Workshop on Humanities in the Semantic Web (WHiSe)*, pages 53–64, 2016.

[58] Ignacio Gutiérrez, V\'\ictor Álvarez, M Paule, Juan Ramón Pérez-Pérez, and Sara de Freitas. Adaptation in E-Learning Content Specifications with Dynamic Sharable Objects. *Systems*, 4(2):24, 2016.

[59] Alon Y. Halevy. Answering queries using views: A survey. *The VLDB Journal*, 10(4):270–294, Dec 2001.

[60] Lushan Han, Tim Finin, Cynthia Sims Parr, Joel Sachs, and Anupam Joshi. RDF123: from spreadsheets to RDF. In *The Semantic Web - ISWC 2008, 7th International Semantic Web Conference, ISWC 2008, Karlsruhe, Germany, October 26-30, 2008. Proceedings*, pages 451–466, 2008.

[61] Stefan Hanenberg. Faith, hope, and love: an essay on software science's neglect of human factors. In William R. Cook, Siobhán Clarke, and Martin C. Rinard, editors, *Proceedings of the 25th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications, OOPSLA 2010, October 17-21, 2010, Reno/-Tahoe, Nevada, USA*, pages 933–946. ACM, 2010.

[62] Michael E. Hansen, Andrew Lumsdaine, and Robert L. Goldstone. Cognitive architectures: a way forward for the psychology of programming. In Gary T. Leavens and Jonathan Edwards, editors, *ACM Symposium on New Ideas in Programming and Reflections on Software, Onward! 2012, part of SPLASH '12, Tucson, AZ, USA, October 21-26, 2012*, pages 27–38. ACM, 2012.

[63] Matthias Hert, Gerald Reif, and Harald C Gall. A comparison of rdb-tordf mapping languages. In *Proceedings of the 7th International Conference on Semantic Systems*, pages 25–32. ACM, 2011.

[64] Pieter Heyvaert, Anastasia Dimou, Aron-Levi Herregodts, Ruben Verborgh, Dimitri Schuurman, Erik Mannens, and Rik Van de Walle. Rmleditor: A graph-based mapping editor for linked data mappings. In *The Semantic Web. Latest Advances and New Domains - 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 - June 2, 2016, Proceedings*, pages 709–723, 2016.

[65] Pieter Heyvaert, Ben De Meester, Anastasia Dimou, and Ruben Verborgh. Declarative rules for linked data generation at your fingertips! In

The Semantic Web: ESWC 2018 Satellite Events - ESWC 2018 Satellite Events, Heraklion, Crete, Greece, June 3-7, 2018, Revised Selected Papers, pages 213–217, 2018.

[66] Rick Jelliffe. The Schematron: An XML structure validation language using patterns in trees. `http://xml.ascc.net/resource/schematron/schematron.html`, 2001.

[67] Hannes Kaufmann and Dieter Schmalstieg. Mathematics and geometry education with collaborative augmented reality. *Computers & Graphics*, 27(3):339–345, 2003.

[68] Holger Knublauch. SPIN - Modeling Vocabulary. `http://www.w3.org/Submission/spin-modeling/`, 2011.

[69] Holger Knublauch and Dimitris Kontokostas. Shapes constraint language (SHACL). https://www.w3.org/TR/shacl/, jun 2017.

[70] Nassim Kobeissy, Marc Girod Genet, and Djamal Zeghlache. Mapping XML to OWL for seamless information retrieval in context-aware environments. In Fusun Ozguner, Buyurman Baykal, Ali Akoglu, and Ozgur Ercetin, editors, *IEEE International Conference on Pervasive Services*, pages 349–354, Istanbul, July 2007. IEEE. doi: 10.1109/PERSER.2007.4283938.

[71] Thomas Kurz, Georg Güntner, Violeta Damjanovic, Sebastian Schaffert, and Manuel Fernandez. Semantic enhancement for media asset management systems. *Multimedia Tools and Applications*, 70(2):949–975, 2012.

[72] Jose Emilio Labra Gayo, Eric Prud'hommeaux, Iovka Boneva, and Dimitri Kontokostas. *Validating RDF Data*, volume 7 of *Synthesis Lectures on the Semantic Web: Theory and Technology*. Morgan and Claypool Publishers, 2018. doi: 10.2200/S00786ED1V01Y201707WBE016.

[73] Damien Lacoste, Kiran Prakash Sawant, and Suman Roy. An efficient XML to OWL converter. In Arun Bahulkar, K. Kesavasamy, T. V. Prabhakar, and Gautam Shroff, editors, *Proceedings of the 4th India software engineering conference*, pages 145–154, Thiruvananthapuram, 2011. ACM. doi: 10.1145/1953355.1953376.

[74] Dongwon Lee, Murali Mani, and Wesley W Chu. Schema Conversion Methods between XML and Relational Models. In Michel Klien and Borys Omelayenko, editors, *Knowledge Transformation for the Semantic Web*, volume 95 of *Frontiers in Artificial Intelligence and Applications*, chapter 1, pages 1–17. IOS Press, 2003.

[75] Maxime Lefrançois, Antoine Zimmermann, and Noorani Bakerally. Flexible RDF generation from RDF and heterogeneous data sources with sparql-generate. In *Knowledge Engineering and Knowledge Management - EKAW 2016 Satellite Events, EKM and Drift-an-LOD, Bologna, Italy, November 19-23, 2016, Revised Selected Papers*, pages 131–135, 2016.

[76] Maxime Lefrançois, Antoine Zimmermann, and Noorani Bakerally. A SPARQL extension for generating RDF from heterogeneous formats. In *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, pages 35–50, 2017.

[77] Dunia Llanes-Padrón and Juan-Antonio Pastor-Sanchez. Records in contexts: the road of archives to semantic interoperability. *Program*, 51(4):387–405, 2017.

[78] Steffen Lohmann, Philipp Heim, and Paloma Díaz. Exploiting the Semantic Web for Interactive Relationship Discovery in Technology Enhanced Learning. In *2010 10th IEEE International Conference on Advanced Learning Technologies*, pages 302–306. IEEE, jul 2010.

[79] Massimo Loi and Bruno Ronsivalle. A Particular Aspect of Cost Analysis in Distance Education. In *Distance and E-Learning in Transition*, pages 161–168. John Wiley & Sons, Inc., 2013.

[80] Patrick Maier, Marcus Tönnis, and Gudron Klinker. Augmented Reality for teaching spatial relations. In *Conference of the International Journal of Arts & Sciences, Toronto*, 2009.

[81] P. Markellou, I. Mousourouli, S. Spiros, and A. Tsakalidis. Using Semantic Web Mining Technologies for Personalized e-Learning Experiences. In *Proceedings of the web-based education*, pages 461–826, Grindelwald, Switzerland, feb 2005. ACTA Press.

[82] Andrew McAfee, Erik Brynjolfsson, Thomas H Davenport, DJ Patil, and Dominic Barton. Big data: the management revolution. *Harvard business review*, 90(10):60–68, 2012.

[83] John McCarthy. What is artificial intelligence? 1998.

[84] Ben De Meester, Pieter Heyvaert, Ruben Verborgh, and Anastasia Dimou. Mapping languages: Analysis of comparative characteristics. In David Chaves-Fraga, Pieter Heyvaert, Freddy Priyatna, Juan F. Sequeda, Anastasia Dimou, Hajira Jabeen, Damien Graux, Gezim Sejdiu, Mohammed Saleem, and Jens Lehmann, editors, *Joint Proceedings of the 1st International Workshop on Knowledge Graph Building and 1st International Workshop on Large Scale RDF Analytics co-located with 16th Extended Semantic Web Conference (ESWC 2019), Portorož, Slovenia, June 3, 2019*, volume 2489 of *CEUR Workshop Proceedings*, pages 37–45. CEUR-WS.org, 2019.

[85] Ben De Meester, Wouter Maroy, Anastasia Dimou, Ruben Verborgh, and Erik Mannens. RML and fno: Shaping dbpedia declaratively. In *The Semantic Web: ESWC 2017 Satellite Events - ESWC 2017 Satellite Events, Portorož, Slovenia, May 28 - June 1, 2017, Revised Selected Papers*, pages 172–177, 2017.

[86] Sergei Melnik and Stefan Decker. Representing Order in RDF. `http://infolab.stanford.edu/~stefan/daml/order.html`, January 2001.

[87] Albert Meroño-Peñuela. Semantic web for the humanities. In Philipp Cimiano, Oscar Corcho, Valentina Presutti, Laura Hollink, and Sebastian Rudolph, editors, *The Semantic Web: Semantics and Big Data: 10th International Conference, ESWC 2013*, volume 7882 of *Lecture Notes in Computer Science*, pages 645–649, Montpellier, May 2013. Springer Berlin Heidelberg. doi: 10.1007/978-3-642-38288-8_44.

[88] Albert Meroño-Peñuela, Ashkan Ashkpour, Marieke van Erp, Kees Mandemakers, Leen Breure, Andrea Scharnhorst, Stefan Schlobach, and Frank van Harmelen. Semantic technologies for historical research: A survey. *Semantic Web*, 6(6):539–564, 2015.

[89] Franck Michel, Loïc Djimenou, Catherine Faron-Zucker, and Johan Montagnat. Translation of relational and non-relational databases into RDF with xr2rml. In Valérie Monfort, Karl-Heinz Krempels, Tim A. Majchrzak, and Ziga Turk, editors, *WEBIST 2015 - Proceedings of the 11th International Conference on Web Information Systems and Technologies, Lisbon, Portugal, 20-22 May, 2015*, pages 443–454. SciTePress, 2015.

[90] Franck Michel, Johan Montagnat, and Catherine Faron Zucker. *A survey of RDB to RDF translation approaches and tools.* PhD thesis, I3S, 2014.

[91] Igor Miletic, Marko Vujasinovic, Nenad Ivezic, and Zoran Marjanovic. Enabling Semantic Mediation for Business Applications: XML-RDF, RDF-XML and XSD-RDFS transformations. In Ricardo J Gonçalves, Jörg P Müller, Kai Mertins, and Martin Zelm, editors, *Enterprise Interoperability II: New Challenges and Approaches*, pages 483–494. Springer London, London, 2007.

[92] Igor Miletic, Marko Vujasinovic, Nenad Ivezic, and Zoran Marjanovic. Enabling Semantic Mediation for Business Applications: XML-RDF, RDF-XML and XSD-RDFS transformations. In Ricardo J Gonçalves, Jörg P Müller, Kai Mertins, and Martin Zelm, editors, *Enterprise Interoperability II: New Challenges and Approaches*, pages 483–494, Madeira, 2007. Springer London. doi: 10.1007/978-1-84628-858-6_53.

[93] Alejandro Montes García, Paul De Bra, George H L Fletcher, and Mykola Pechenizkiy. A DSL Based on CSS for Hypertext Adaptation. In *Proceedings of the 25th ACM Conference on Hypertext and Social Media*, HT '14, pages 313–315, New York, NY, USA, 2014. ACM.

[94] Adriaan Moors, Frank Piessens, and Martin Odersky. Parser combinators in Scala. *Department of Computer Science, KU Leuven*, 2008. `https://limo.libis.be/primo-explore/fulldisplay?docid=LIRIAS1652814&context=L&vid=Lirias&search_scope=Lirias&tab=default_tab&lang=en_US`.

[95] Heiko Müller, Liliana Cabral, Ahsan Morshed, and Yanfeng Shu. From restful to SPARQL: A case study on generating semantic sensor data. In *Proceedings of the 6th International Workshop on Semantic Sensor Networks co-located with the 12th International Semantic Web Conference (ISWC 2013), Sydney, Australia, October 22nd, 2013.*, pages 51–66, 2013.

[96] Piedra Nelson, Tovar Edmundo, Colomo-Palacios Ricardo, Lopez-Vargas Jorge, and Alexandra Chicaiza Janneth. Consuming and producing linked open data: the case of OpenCourseWare. *Program*, 48(1):16–40, 2014.

[97] Falco Nogatz and Thom Frühwirth. *From XML Schema to JSON Schema - Comparison and Translation with Constraint Handling Rules*. Bachelor Thesis. Ulm: University of Ulm, Ulm, Germany, 2013.

[98] Martin J. O'Connor and Amar Das. Acquiring OWL ontologies from XML documents. In Mark A. Musen and Oscar Corcho, editors, *Proceedings of the sixth international conference on Knowledge capture*, pages 17–24, Banff, June 2011. ACM. doi: 10.1145/1999676.1999681.

[99] Sung Youl Park. An Analysis of the Technology Acceptance Model in Understanding University Students' Behavioral Intention to Use e-Learning. *Journal of Educational Technology & Society*, 12(3):150–162, 2009.

[100] MPuerto Paule-Ruiz, Víctor Álvarez-García, J R Pérez-Pérez, and M Riestra-González. Voice Interactive Learning: A Framework and Evaluation. In *Proceedings of the 18th ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '13, pages 34–39, New York, NY, USA, 2013. ACM.

[101] Giuseppe Della Penna, Antinisca Di Marco, Benedetto Intrigila, Igor Melatti, and Alfonso Pierantonio. Interoperability mapping from XML Schemas to ER diagrams. *Data & Knowledge Engineering*, 59(1):166–188, 2006. doi: 10.1016/j.datak.2005.08.002.

[102] Silvio Peroni and David Shotton. The spar ontologies. In *The Semantic Web – ISWC 2018*, pages 119–136, Cham, 2018. Springer International Publishing.

[103] Christopher Pollin and Georg Vogeler. Semantically enriched historical data. drawing on the example of the digital edition of the "urfehdebucher der stadt basel". In *Second Workshop on Humanities in the Semantic Web (WHiSe)*, 2017.

[104] Eric Prud'hommeaux, Iovka Boneva, Jose Emilio Labra Gayo, and Gregg Kellogg. Shape expressions language 2.0. `http://shex.io/shex-semantics/index.html`, 2017.

[105] Eric Prud'hommeaux, Jose Emilio Labra Gayo, and Harold Solbrig. Shape Expressions: An RDF Validation and Transformation Language. In *Proceedings of the 10th International Conference on Semantic Systems*, SEM '14, pages 32–40, New York, NY, USA, 2014. ACM.

[106] Eric Prud'hommeaux, Jose Emilio Labra Gayo, and Harold Solbrig. Shape expressions: an RDF validation and transformation language. In Harald Sack, Agata Filipowska, Jens Lehmann, and Sebastian Hellmann, editors, *Proceedings of the 10th International Conference on Semantic Systems, SEMANTiCS 2014*, pages 32–40, Leipzig, September 2014. ACM. doi: 10.1145/2660517.2660523.

[107] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016.

[108] David Reinsel, John Gantz, and John Rydning. The Digitization of the World. From Edge to Core. Technical report, Seagate, IDC, November 2018. Last accessed: October 28, 2019.

[109] Toni Rodrigues, Pedro Rosa, and Jorge Cardoso. Mapping XML Schema to Existing OWL ontologies. In Pedro Isaías, Miguel Baptista Nunes, and Inmaculada J. Martínez, editors, *International Conference WWW/Internet*, volume II, pages 72–77, Murcia, October 2006. IADIS.

[110] Satya S Sahoo, Wolfgang Halb, Sebastian Hellmann, Kingsley Idehen, Ted Thibodeau Jr, Sören Auer, Juan Sequeda, and Ahmed Ezzat. A survey of current approaches for mapping of relational databases to rdf. *W3C RDB2RDF Incubator Group Report*, 1:113–130, 2009.

[111] François Scharffe, Laurent Bihanic, Gabriel Képéklian, Ghislain Atemezing, Raphaël Troncy, Franck Cotton, Fabien Gandon, Serena Villata, Jérôme Euzenat, Zhengjie Fan, Bénédicte Bucher, Fayçal Hamdi, Pierre-Yves Vandenbussche, and Bernard Vatant. Enabling linked data publication with the datalift platform. In *Semantic Cities, Papers from the 2012 AAAI Workshop, Toronto, Ontario, Canada, July 22-23, 2012.*, 2012.

[112] Susan Schreibman, Ray Siemens, and John Unsworth. *A companion to digital humanities*. John Wiley & Sons, 2008.

[113] Juan F. Sequeda, Marcelo Arenas, and Daniel P. Miranker. On directly mapping relational databases to RDF and OWL. In *Proceedings of the 21st World Wide Web Conference 2012, WWW 2012, Lyon, France, April 16-20, 2012*, pages 649–658, 2012.

[114] Charles Severance, Ted Hanss, and Josepth Hardin. Ims learning tools interoperability: Enabling a mash-up approach to teaching and learning tools. *Technology, Instruction, Cognition and Learning*, 7(3-4):245–262, 2010.

[115] John Simpson and Susan Brown. From XML to RDF in the Orlando Project. In Kozaburo Hachimura, Toru Ishida, Naoko Tosa, Donghui Lin, and Akira Maeda, editors, *2013 International Conference on Culture and Computing*, pages 194–195, Kyoto, September 2013. IEEE. doi: 10.1109/CultureComputing.2013.61.

[116] Jason Slepicka, Chengye Yin, Pedro A. Szekely, and Craig A. Knoblock. KR2RML: an alternative interpretation of R2RML for heterogenous sources. In Olaf Hartig, Juan F. Sequeda, and Aidan Hogan, editors, *Proceedings of the 6th International Workshop on Consuming Linked Data co-located with 14th International Semantic Web Conference (ISWC 2105), Bethlehem, Pennsylvania, USA, October 12th, 2015*, volume 1426 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.

[117] C Michael Sperberg-McQueen and Eric Miller. On mapping from colloquial XML to RDF using XSLT. In *Extreme Markup Languages®*, 2004.

[118] C. Michael Sperberg-McQueen and Eric Miller. On mapping from colloquial XML to RDF using XSLT. In *Proceedings of Extreme Markup Languages® 2004*, Montreal, 2004. http://conferences.idealliance.org/extreme/html/2004/Sperberg-McQueen01/EML2004Sperberg-McQueen01.html.

[119] Claus Stadler, Jörg Unbehauen, Patrick Westphal, Mohamed Ahmed Sherif, and Jens Lehmann. Simplified RDB2RDF mapping. In Christian Bizer, Sören Auer, Tim Berners-Lee, and Tom Heath, editors, *Proceedings of the Workshop on Linked Data on the Web, LDOW 2015, co-located with the 24th International World Wide Web Conference (WWW 2015), Florence, Italy, May 19th, 2015*, volume 1409 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.

[120] L Stojanovic, S Staab, and R Studer. eLearning Based on the Semantic Web. In *Proceedings of WebNet 2001, World Conference on the WWW and the Internet*, number FEBRUARY 2002, Orlando, Florida, 2001.

[121] Timo Sztyler, Jakob Huber, Jan Noessner, Jaimie Murdock, Colin Allen, and Mathias Niepert. LODE: Linking digital humanities content to the web of data. In George Buchanan, Martin Klein, Andreas Rauber, and Sally Jo Cunningham, editors, *IEEE/ACM Joint Conference on Digital Libraries*, pages 423–424, London, September 2014. IEEE and ACM. doi: 10.1109/JCDL.2014.6970206.

[122] Jeremy Tandy, Ivan Herman, and Gregg Kellogg. Generating rdf from tabular data on the web, w3c recommendation 17 december 2015. w3c recommendation. https://www. w3. org/TR/2015/REC-csv2rdf-20151217, 2015.

[123] Jiao Tao, Evren Sirin, Jie Bao, and Deborah L. McGuinness. Integrity constraints in OWL. In Maria Fox and David Poole, editors, *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI-10)*, pages 1443–1448, Atlanta, July 2010. AAAI.

[124] Stamatios Theocharis and George A. Tsihrintzis. Rdf serialization from JSON data: The case of JSON data in diavgeia.gov.gr. In *7th International Conference on Information, Intelligence, Systems & Applications, IISA 2016, Chalkidiki, Greece, July 13-15, 2016*, pages 1–6, 2016.

[125] Pham Thi Thu Thuy, Young-Koo Lee, Sungyoung Lee, and Byeong-Soo Jeong. Transforming valid XML documents into RDF via RDF schema. In *Next Generation Web Services Practices, 2007. NWeSP 2007. Third International Conference on*, pages 35–40. IEEE, 2007.

[126] Pham Thi Thu Thuy, Young-Koo Lee, Sungyoung Lee, and Byeong-Soo Jeong. Transforming valid XML documents into RDF via RDF schema. In Ajith Abraham and Sang Yong Han, editors, *Third International Conference on Next Generation Web Services Practices (NWeSP 2007).*, pages 35–40, Seoul, October 2007. IEEE. doi: 10.1109/NWESP.2007.23.

[127] Pham Thi Thu Thuy, Young-Koo Lee, Sungyoung Lee, and Byeong-Soo Jeong. Exploiting XML schema for interpreting XML documents as RDF.

In *Services Computing, 2008. SCC'08. IEEE International Conference on*, volume 2, pages 555–558. IEEE, 2008.

[128] Pham Thi Thu Thuy, Young-Koo Lee, Sungyoung Lee, and Byeong-Soo Jeong. Exploiting XML schema for interpreting XML documents as RDF. In Wil van der Aalst, Calton Pu, Elisa Bertino, Ephraim Feig, and Patrick C. K. Hung, editors, *2008 IEEE International Conference on Services Computing (SCC'08)*, volume 2, pages 555–558, Honolulu, 2008. IEEE. doi: 10.1109/SCC.2008.93.

[129] Georg Vogeler. Towards a Standard of Encoding Medieval Charters with XML. *Literary and Linguistic Computing*, 20(3):269–280, 09 2005.

[130] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, 2014.

[131] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*, 3(1):1–9, 2016.

[132] Hong Qing Yu, C Pedrinaci, S Dietze, and J Domingue. Using Linked Data to Annotate and Search Educational Video Resources for Supporting Distance Learning. *Learning Technologies, IEEE Transactions on*, 5(2):130–142, 2012.