Preserving the Alignment of LD with Source Data

Alex Randles[1] and Declan O'Sullivan[1]

[1] *ADAPT Centre for Digital Content, Trinity College Dublin, Dublin, Ireland*

Abstract

A significant proportion of Linked data (LD) is created through mapping of data from a variety of sources of data. Linked data has been described as highly dynamic in nature with source data being continuously changed, which could impact the quality of the linked data and related mapping artefacts. Changes which have occurred in the source data of linked data datasets should be propagated into the resulting dataset to provide an accurate representation of the underlying data sources. These changes can occur at an extremely fast rate which can result in difficulties propagating each change in a timely manner. Surprisingly, despite the growth of linked data publication on the web of data, there exists no standard to address the dynamics of the data. An approach which captures changes in the source data used by mapping artefacts to create linked data datasets will help to address the dynamics involved in the publication process. Furthermore, capturing changes in a machine-readable format will allow software agents to automatically process them and take appropriate actions to preserve the alignment between mapping artefacts and data sources used to create the linked data dataset. Moreover, the ability to monitor the source data and detect changes regularly will support a mechanism to automatically send notifications of changes and potential alignment issues to data producers, therefore, providing necessary information to guide them in improving alignment. Evaluating an approach designed to address the dynamics of linked data is important to provide evidence of sufficient usability. This paper describes the evaluation of the Mapping Quality Improvement (MQI) Framework and focuses on change detection of source data used to create linked data and aims to support data producers in providing timely data to consumers and improving the quality, maintenance and reuse of related mapping artefacts. The evaluation of the MQI framework involved 55 participants with varying levels of background knowledge.

Keywords

Usability Testing; Dataset Dynamics; Linked Data; Mappings; Data Quality.

1. Introduction

Declarative uplift mapping artefacts are used to generate linked data datasets and contain rules for converting Non-Resource Description Framework (RDF) data, in formats such as XML, CSV, relational data into a RDF representation [19]. Various representations of these mapping artefacts exist, such as RDB to RDF Mapping Language (R2RML) [3], which is the World Wide Web Consortium (W3C) recommendation for transforming relational data into RDF and allows customized transformation rules to be defined. Another prominent representation is RDF Mapping Language (RML) [4], which extends R2RML to allow more diverse source data formats, such as XML, CSV and JSON. The resulting linked data datasets are highly dynamic in nature with resources continuously being added and removed in an attempt to improve data quality by updating resources and respective vocabularies as they evolve [23]. Oftentimes, the dynamics of the linked data dataset is measured by the ”freshness” quality dimension, which relates to the age and occurrences of changes in data [2] and has been described as one of the most important aspects of linked data quality [2]. Such 'freshness' is crucial to underpin machine learning processes enabling an Internet of Things and People [1]. Interestingly, the issue of detecting and propagating changes in linked data has been discussed for over a decade, however, no defacto approach or standards-based approach exists to tackle the problem [22]. Existing approaches [12,13,21] in the state of the art predominantly propose methods to address the dynamics of resources and interlinks in linked data datasets, however, one approach [24] exists which targets the dynamics of the source data of linked data and focuses on relational data and R2RML mappings. In this paper, an approach is proposed to capture change information in heterogenous formats used to create linked data datasets and allowing these changes to be propagated into the resulting data, with the aim of supporting an increase in linked data dataset freshness [20]. In addition, a notification policy approach is included, which enables data producers to be informed of changes in a timely manner. A usability evaluation has been conducted on the proposed approach in an attempt to validate the design with end users [10]. In addition, usability testing provides an opportunity to support collaboration between domain experts and computer scientists when developing tools and processes [10]. Characterizing respective end users based on background knowledge, allows the level of knowledge to sufficiently use the tool to be determined [19].

In this paper we discuss the design and evaluation of the second iteration of the **MQI Framework** [15,19,20]. The first iteration of the framework included a component designed to assess and refine the quality of R2RML mappings. The component uses the **Mapping Quality Improvement Ontology (MQIO)**[1] [16,17] to represent captured mapping quality information in RDF format. The second iteration of the framework includes the original functionality and is extended to include a component for change detection of source data, represented in heterogeneous formats. In addition, the component detects links between detected changes and respective mappings. Changes which are captured by this component are represented in the **Ontology for Source Change Detection (OSCD)**[2] [20]. We also describe an extension to the functionality of the framework to provide suggestions to agents on how to improve alignment. The objective of the framework is to improve the quality of mappings, while preserving alignment with underlying data sources, with the aim of providing fresh data to consumers. The remainder of this paper is structured as follows: **Section 2** discusses the design of MQI framework, including the utilization of OSCD. **Section 3** outlines additional functionality integrated into the framework as a result of the evaluation. **Section 4** describes the evaluation setup and results. **Section 5** discusses related work in the state of the art. **Section 6** outlines future work and concludes the paper.

2. Assessing LD Alignment

The first iteration of the MQI framework [18,19] included a component to assess and refine the quality of R2RML [3] mappings involved in the generation of linked data datasets. The second iteration of MQI extends the original functionality to add a component to detect source data changes and link them with respective mappings. In addition, support for RML [4] mapping artefacts, is included, allowing source data represented in heterogenous formats to be used as input to the framework. The detected changes are represented according to OSCD [20], which was previously developed by the authors of this paper in order to model information related to source data changes. The ontology is utilized by the framework to represent and interchange information related to changes detected in source data. The ontology was designed as format independent and can be used to represent detected changes in source data formats such as XML, CSV, JSON, relational data, among others. The ontology can be used to model changes in source data formats supported by R2RML and RML. In addition, OSCD enables notification policies to be defined using the Rei policy ontology [7], which provides mapping engineers with timely information on detected changes and their associated potential alignment issues in the resulting linked data dataset. Representing changes in OSCD allows them to be linked with the mapping artefact itself and associated quality reports, such as those represented in MQIO [16,17], as a result of the mapping quality assessment and improvement component of the framework. **Figure 1** presents a diagram of the source data change detection component of the MQI framework.

[1] MQIO specification at https://w3id.org/MQIO
[2] OSCD specification at https://w3id.org/OSCD

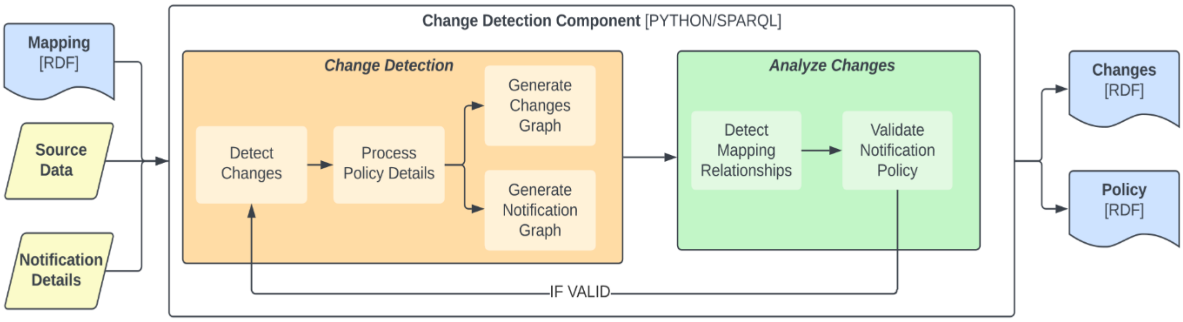


Figure 1: Overview of processes involved in the change detection component of the MQI Framework

The process is outlined below.

- **Input:** Two versions of source data and respective mappings represented in RML or R2RML are inputted into the framework. Oftentimes, mappings will have been previously uploaded to the mapping quality assessment and refinement component of the MQI framework. In addition, notification details can be input in order to create a policy which defines when users will be notified of detected changes in the source data.
- **Change Detection:** Changes are detected between the versions using existing methods, such as file comparison. Thereafter, the detected changes (and the notification details input into framework) are uplifted in RDF format, resulting in two named graphs.
- **Analyze Changes:** The detected changes are linked with the inputted mapping artefacts in order to identify changes which could impact them, for example a data reference in a mapping that does not exist in the current source data.
- **Output:** The resulting two named graphs detail detected source data changes and notification policy. The changes are periodically detected until the notification policy becomes invalid by fulfillment of a change threshold or end date.

The MQI framework is implemented using the following technologies.

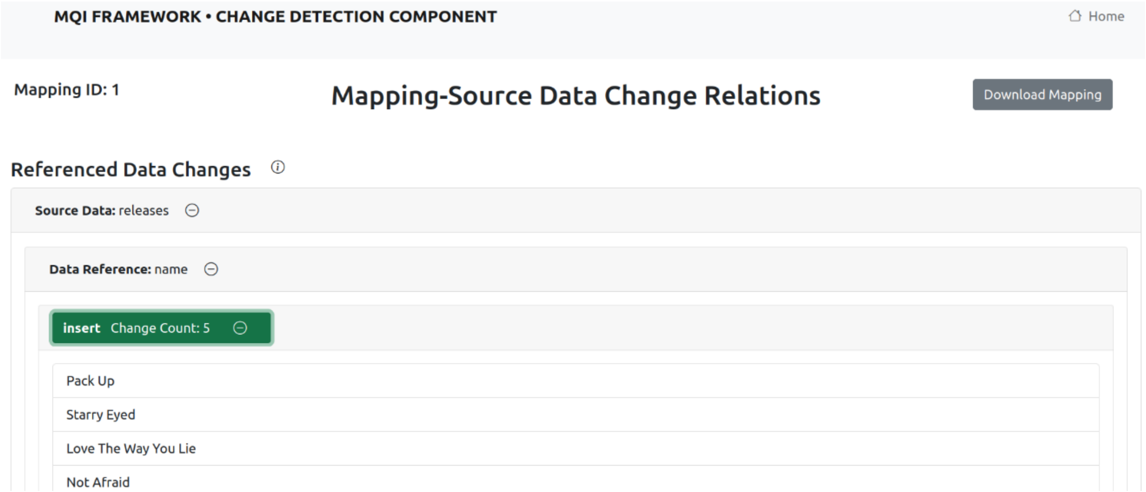
- Several **Python** libraries are used in the implementation. The Flask library [5] was used to create a web application with a Graphical User Interface (GUI). The RDFLib library [9] enables execution of SPARQL queries and is used to query and update RDF data. A number of file comparison methods are used. XMLDiff[3] is used to compare XML files. CSVDiff[4] is used to compare CSV files. MySQL[5] library is used to compare relational data.
- **SPARQL** [6] is used to link graphs containing detected changes and associated notification policy, with respective mapping artefacts.
- **R2RML** [3] is used to uplift information captured by the MQI framework, that is detected changes and notification details, into RDF format.

Detecting changes using the implementation involves the following steps.

1. The versions of source data input into the GUI are compared using one of the aforementioned methods and the result stored.
2. The results are uplifted into RDF using an R2RML mapping expressed according to the OSCD (see next section).
3. Input notification details are uplifted using an R2RML mapping expressed according to the Rei policy ontology [7].
4. SPARQL queries are used to retrieve necessary information in order to provide an overview of the detected changes to users and link changes[6] with respective mappings in order to identify potential alignment issues.

[3] https://pypi.org/project/xml-diff/
[4] https://pypi.org/project/csv-diff/
[5] https://pypi.org/project/mysql-connector-python/
[6] https://raw.githubusercontent.com/alex-randles/KGCW-2023-Supplementary/main/linking_query.rq

Figure 2 presents a screenshot of the implementation displaying drop-down information detailing links between a sample source data and its mapping artefact.



MQI FRAMEWORK • CHANGE DETECTION COMPONENT

Mapping ID: 1

Mapping-Source Data Change Relations

Referenced Data Changes

Source Data: releases

Data Reference: name

insert Change Count: 5

Pack Up

Starry Eyed

Love The Way You Lie

Not Afraid

Figure 2: Screenshot of MQI framework displaying source data changes related to a mapping

The MusicBrainz project[7] involves an online music encyclopedia, which contains music metadata, such as artist, labels, recordings and releases. The project has created 12 R2RML [3] mappings designed to uplift information in the encyclopedia into linked data representation and one of them is designed to transform the releases of artists. The mappings[8] source data is a table ("releases") in a relational database and contains two term maps to map the ID ("gid") and name ("name") of the release. For instance, 5 releases have been added and detected by the framework, which should be propagated into the resulting dataset in order to preserve the freshness of the data [20]. The screenshot shows the name of the releases which have been added to the source data. It is difficult to determine when the mapping should be regenerated as releases could be added frequently or infrequently. Therefore, a notification policy should be defined to ensure timely updates of relevant information, which provides an indication of when the mapping should be regenerated in order to capture new releases.

3. Improving LD Alignment

Additional functionality has been added to the framework since the conclusion of the evaluation described in this paper. The functionality is designed to automatically suggest actions which could be executed to improve the level of alignment between mappings and respective source data. In addition, Shapes Constraint Language (SHACL) [8] constraints are proposed to assess the level of alignment.

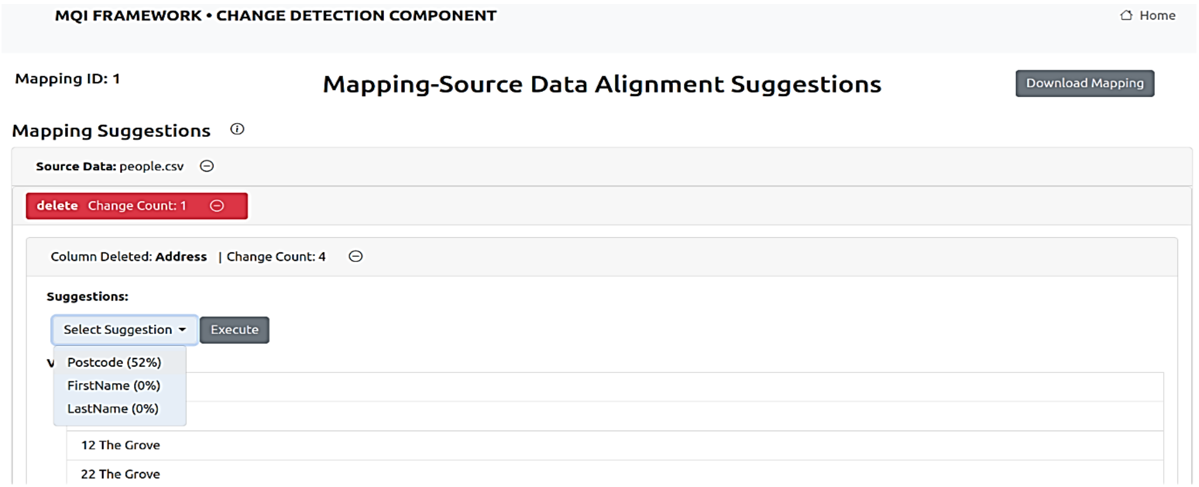
3.1. Alignment of Source Changes with Mapping

Figure 3 presents a screenshot of the MQI framework displaying mapping suggestions to improve alignment for sample source data[9], which contained information about people, including their name and address.

[7] https://musicbrainz.org/

[8] https://raw.githubusercontent.com/metabrainz/MusicBrainz-R2RML/master/mappings/release.ttl

[9] https://raw.githubusercontent.com/alex-randles/KGCW-2023-Supplementary/main/people.csv



Mapping ID: 1

Mapping-Source Data Alignment Suggestions

Source Data: people.csv

Change Count: 1

Column Deleted: Address | Change Count: 4

Suggestions:

12 The Grove

22 The Grove

Figure 3: Screenshot of MQI framework displaying suggestions to improve alignment

The sample source data ("people.csv") has had a column referenced in a related mapping[10] removed, therefore, the mapping is no longer compatible with the current source data. The column ("Address") removed contained data on the location of people, however, a column ("Postcode") containing their postcodes ("Change Count: 4") has been inserted. The framework compares previous columns with the names of the current columns to identify indications that they are related. The comparison is completed using WordNet Similarity[11], which is software designed to measure semantic similarity between a pair of concepts. The similarity score for the "Address" and "Postcode" is 52%, which indicated they have similarities. The framework will provide a suggestion in this case as the score is above the threshold (> 0.25). Thereafter, the framework will automatically update the mapping by executing a SPARQL query[12] if the suggestion is accepted by the user.

3.2. SHACL Shapes

SHACL [8] is a W3C recommendation designed to validate the quality of RDF graphs, which can be applied to mappings represented in RDF format, such as R2RML [3] and RML [4]. Shapes refer to constraints defined using the properties and classes in the SHACL vocabulary. The functionality to generate SHACL shapes from the original source data has been integrated into the MQI framework. The shapes can be applied to mappings at any point during their evolution in order to easily allow the identification of alignment issues with underlying data sources. The framework generates a shape which validates if each data reference in the mapping is in the source data. **Table 1** presents the pseudocode used to generate the shapes and the resulting shape for the RML mapping used in the evaluation.

Table 1: Pseudocode for Shape Generation (A) and SHACL Shape generated (B)

	A	B

1	***Input:*** *columns of original source data*	`schema:PersonShape`
		`  a sh:NodeShape ;`
		`  sh:targetObjectsOf  rr:objectMap ;`
		`  sh:property [`
		`    sh:path rr:column, rml:reference;`
		`    sh:in ("ID" Address");`
		`    sh:message "Data reference no longer`
		`      in source data." ;`
		`  ] .`

2	***column-count*** *← count total number of columns*	
3	***Output:*** *SHACL shape to assess alignment*	
4	***Initialization of Variables:*** *Assign zero to variable* ***i*** *and empty list to* ***columns***	
5	***while (i < column-count) do*** *// Iterate column names*	
6	***column-name*** *← retrieve current column name using* ***i***	
7	*append* ***column-name*** *to* ***columns*** *list*	
8	***end***	
9	*compute remaining shape targeting* ***rr:objectMap***	
10	*compute SHACL list using* ***columns, rr:column*** *and* ***rml:reference***	

[10] https://raw.githubusercontent.com/alex-randles/KGCW-2023-Supplementary/main/sample_mapping.ttl
[11] https://www.nltk.org/howto/wordnet.html
[12] https://raw.githubusercontent.com/alex-randles/KGCW-2023-Supplementary/main/update_query.rq

Pseudocode (**A**) is shown which outlines the process involved in generating a SHACL shape (**B**) from the RML mapping. The same process can be applied to R2RML mappings and involves adding each attribute (e.g. column, element, row) name in the original source data into a SHACL list (`sh:in`), which can be used to validate that the attribute exists in the current source data. In this case, the sample mapping will no longer be compatible with the source data as the "Address" column has been changed to "Postcode", which should be updated accordingly. The following SHACL validation report[13] (**Listing 1**) will be generated when the shape shown is executed on the sample mapping.

```
[   a sh:ValidationReport ;
sh:conforms false ;
sh:result [
    a sh:ValidationResult ;
    sh:resultSeverity sh:Violation ;
    sh:sourceConstraintComponent sh:InConstraintComponent ;
    sh:sourceShape _:n839 ;
    sh:focusNode _:n959 ;
    sh:value "Address" ;
    sh:resultPath rml:reference;
    sh:resultMessage "Data reference no longer in source data" ; ] .
```

Listing 1: SHACL validation report generated when sample shape executed

The SHACL validation report (`sh:ValidationReport`) is expressed in the SHACL validation report vocabulary[14]. The report shown includes 1 violation (`sh:ValidationResult`), which has detected a column (`sh:value`) is no longer present in the source data of the mapping (`sh:message`). The validation report is machine-readable and queryable by SPARQL [6], as it is represented in RDF format, which can be used to automatically update the mapping in order to preserve alignment.

4. Evaluation

The following section describes the user evaluation conducted on the change detection component of the MQI framework. Firstly, the methodology and metrics used in the study are discussed. Thereafter, the results of the study are described. Finally, a discussion on the hypotheses is presented. The hypotheses related to this study were:

- **H1)** The framework facilitates the identification of changes in source data and links with respective mappings;
- **H2)** The participants' background knowledge influences the successful completion of the tasks.

The hypotheses were defined to allow measurement of required level of knowledge to successfully interact with the framework,

4.1. Methodology

A user evaluation was conducted to test the hypotheses related to this study. The participants were grouped into two cohorts, **student** and **expert** cohort, depending on level of background knowledge. The participants were provided with sample source data and a related mapping, which would allow them to interact with the framework, in order to identify source data changes and links with mappings. Hypothesis **H1** was tested by analyzing the results of each cohort for the Understanding Questionnaire and the Post Study Usability Questionnaire (PSSUQ). Hypothesis **H2** was tested by comparing the results of these questionnaires for both cohorts.

[13] https://raw.githubusercontent.com/alex-randles/KGCW-2023-Supplementary/main/shacl_report.ttl

[14] https://www.w3.org/TR/shacl/#dfn-validation-report-vocabulary

4.2. Metrics

The metrics used for the evaluation included the scores and comments of three questionnaires and qualitative data analysis from the data captured in the questionnaires.

Post Study Usability Questionnaire (PSSUQ). The PSSUQ [10] is a standardized questionnaire which measures the satisfaction provided by a piece of software to users. The questionnaire was developed by IBM and has had extensive psychometric evaluation completed on it, unlike similar questionnaires such as the System Usability Scale (SUS)[15]. The questionnaire consists of 19 positive statements related to the satisfaction of software and are scored on a Likert scale from 1 (Best Case) – 7 (Worst Case). In addition, an open comment section accompanies each question. Four metrics are measured by the PSSUQ which include system usefulness (SysUse), information quality (InfoQual), interface quality (IntQual) and Overall.

Understanding Questionnaire. A questionnaire[16] (**Table 2**) was created to test if participants could understand the change detection information provided by the MQI framework. The questionnaire included two sections which related to the change detection processes (Section 1) and changes which have been detected in the source data and links with respective mappings (Section 2).

Table 2: Understanding Questionnaire used in evaluation

#	Section 1	Section 2
1	How many total changes were detected between the source data files?	A "Referenced Data Change" is one of the following:
2	How many mappings were impacted from the source data changes?	How many columns have been inserted in the source data?
3	A threshold is one of the following:	Select two values which have been inserted into the "FirstName" column in the source data.
4	How many total changes were included in the thresholds?	How many columns have been deleted in the source data?
5	What is the threshold for insert changes?	Which column has been deleted in the source data?
6	Select the two data references in Mapping #1:	How many total values have been inserted into the "ID" data reference?

The questions in Section 1 (S1) were designed to request information related to the total number of changes detected in the source data (S1.Q1), notification policy details (S1.Q3-5), and related mapping details (S1.Q6). The questions in Section 2 (S2) were designed to request information related to types of changes detected (S2.Q1) and other details about them, such as location and number of values changed (S2.Q2-6).

Ontology Application Questionnaire. In addition, a questionnaire was created to ask for feedback from participants in the expert cohort on the application of OSCD in the graph used during the experiment (**Table 3**). The student cohort were not asked for feedback on the application as they have limited ontology design knowledge.

Table 3: Questions on the application of OSCD in the graph used in evaluation

#	Question
1	Do you think the OSCD should be altered to include new concepts/relationships?
2	Do you think the graph of changes detected generated by the application based on the OSCD (as a vocabulary) could be better organized or presented to the user?
3	Any additional comments?

It was hoped the questionnaire would allow feedback to be gathered related to the developed ontology (Q1) and the application of OSCD (Q2) within the graph used in the experiment. In addition, the open comment question (Q3) allowed additional diverse feedback to be gathered.

Thematic Analysis. Thematic analysis [11] is a qualitative data analysis method used to identify patterns. The method involves deriving themes from data which represent discovered patterns. The themes consist of codes which relate to a specific area within the design of a software tool. The analysis

[15] https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html

[16] Complete questionnaire available at https://forms.gle/oLCRyXZQQsEmfMis6

was completed on the qualitative data collected in the open comment sections of the PSSUQ. The process involved the following six-steps, which includes 1) Familiarizing yourself with the data 2) Generation of initial codes 3) Searching for themes 4) Reviewing themes 5) Defining and naming themes and 6) Producing the report. The themes and codes were iteratively refined during the analysis.

4.3. Experiment Setup

The following section discusses the participants involved in the evaluation and tasks which they were asked to complete.

Sample Size. Participants in the **student** cohort have little experience of the mapping process involved in creating linked data datasets. These participants have little experience in creating and operating mappings, however, they have a basic knowledge of semantic web technologies, such as RDF and R2RML. Participants in the **expert** cohort are researchers who are very knowledgeable with RDF and related mapping languages. These participants have experience in creating and operating mappings in a research environment. 48 students were initially recruited, which was reduced to 45 participants after inclusion/exclusion criteria was applied. The expert cohort consisted of 10 participants.

Tasks. The tasks[17] to be undertaken by participants were designed to evaluate the main characteristics of the source data change detection component of the MQI framework. Tasks 1-2 involved the quality assessment of the mapping related to the source data. Tasks 3-7 involved initiation of the change detection process on the source data. Task 8 involved the examination of an overview of the change detection processes. Task 9 and 10 only applied to the expert cohort. The two tasks were designed to retrieve expert feedback on the application of OSCD within the graphs generated. As previously stated, the participants in the student cohort were not asked for feedback as their knowledge of ontology design and application are limited. Task 11-12 involved the examination of detected links between changes in the source data and mapping. Task 13 involved the completion of the questionnaires which measured perceived satisfaction and understanding.

4.4. Experiment Data

The data provided to the participants consisted of three items: 1) RML [4] mapping; 2) Original source data; and 3) Changed source data. The original source data and mapping were retrieved from the RML test case files[18]. The data contains information about famous sports personalities such as their names, a unique identifier (ID), associated sport and place of birth. The changed source data was derived from the test cases and additional similar changes were added by the lead author of this paper. **Listing 2** presents the two versions of the source data used during the experiment.

ID, Name	ID, FirstName, LastName, Sport, City
10, Venus	10, Venus, Williams, Tennis, California 11, Cristiano, Ronaldo, Soccer, Funchal 12, Michael, Jordan, Basketball, Brooklyn 13, Tom, Brady, Football, San Mateo

Listing 2: Original version (left) and Changed version (right) of source data

The RML mapping[19] used during the experiment was designed to uplift the information in the original version of the source data. The name of the "ID" column referenced in the mapping is unchanged between the versions of source data. However, 3 additional values have been added to the column. New columns, "Sport" and "City" have been added with additional data. However, changes between the versions have resulted in the mapping becoming incompatible with the current version of source data, as the "**Name**" column has been split into "FirstName" and "LastName", respectively. Therefore, the alignment between the mapping and source data should be improved to prevent a

[17] https://drive.google.com/file/d/14Rjmi-xVXJ8GJZAdC9uNHxlOP1r8fQa7

[18] https://rml.io/test-cases/

[19] https://raw.githubusercontent.com/kg-construct/rml-test-cases/master/test-cases/RMLTC0002a-CSV/mapping.ttl

decrease in quality [19]. The graph generated by the MQI framework, which contains detected changes during the experiment, expressed in OSCD is available[20].

4.5. Experiment Execution

Participants in both cohorts were informed that assistance was available via email and contact details provided to them.

Completion of Experiment. The participants in both cohorts completed the experiment in an identical structure apart from the questionnaire which included 1 additional section for the expert cohort, which was outlined in **Section 4.2**. First, they were provided with a document[21], which contained information on the following: 1) MQI framework details; 2) Experiment details; and 3) Task sheet. Thereafter, they accessed the framework using the provided details and completed the tasks, including the questionnaire.

Experiment Assistance. None of the participants in either cohort required assistance to complete the tasks involved in the experiment.

4.6. Experiment Results: Student Cohort

The results of the **student** cohort consisted of the scores of the PSSUQ and results from the understanding questionnaire.

4.6.1. PSSUQ Results

Figure 4 presents the PSSUQ scores for each question (Q1-19) and metric of the **student** cohort. A full list of the PSSUQ questions for reference is available[22]. As a note, PSSUQ scores are graded on a scale of 1 (Best case) and 7 (Worst case).

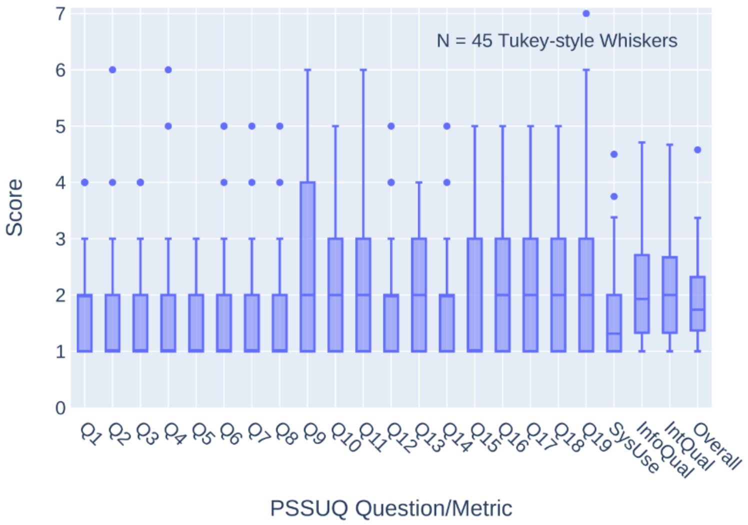


Figure 4: Box plot of PSSUQ scores for **student** cohort

The metric scores were compared with acceptable thresholds found in research [10]. Each metric scored better than its threshold by at least 20%, which included system usefulness (63%), information quality (46%), interface quality (21%) and overall (47%). A third quartile is a statistical measurement in which 75% of the data points are below. Most questions (18 out of 19) had a third quartile of 3 or less. Only one question (Q9) had a third quartile of 4, which related to error messages, however, the

[20] https://raw.githubusercontent.com/alex-randles/KGCW-2023-Supplementary/main/evaluation_graph.ttl
[21] https://drive.google.com/file/d/1pemmMIuW3cLeuSsAnqAKkMLWaeMQ9aqc
[22] https://drive.google.com/file/d/1lIIW23lI3Y25ChxQsVttsT4oRmBWTrSB

question is commonly noted as an outlier as none are shown during most experiments [10]. The overall results of the PSSUQ indicated sufficient satisfaction with the metrics and questions scoring better than their respective thresholds.

4.6.2. Understanding Questionnaire Results

Table 4 presents the scores of the understanding questionnaire for the **student** cohort. The mean score (**μ**) and standard deviation (**σ'**) for each section of the understanding questions are shown.

Table 4: Summary of understanding questions results for **student** cohort

Question #	Section 1 (μ)	Section 1 (σ')	Section 2 (μ)	Section 2 (σ')
Q1	0.93	0.24	0.72	0.46
Q2	0.59	0.5	0.85	0.36
Q3	0.52	0.51	0.98	0.15
Q4	0.85	0.36	0.96	0.21
Q5	0.98	0.15	0.91	0.28
Q6	0.98	0.15	0.91	0.28
Overall (μ)	0.81	0.32	0.89	0.29

Most (9 out of 12) questions had a mean score of at least 80% correct, which indicates overall sufficient understanding from participants of the information presented to them about changes detected. In addition, the low standard deviation of both sections (< 0.35) indicates that the scores are clustered around the mean. However, the worst scoring questions scored (2 out of 12) below 60% and related to information presented on the number of mappings impacted by changes detected and their related thresholds. This will require further clarification, involving the addition of textual descriptions to the interface.

4.7. Experiment Results: Expert Cohort

The results of the **expert** cohort consisted of the PSSUQ scores, scores of the understanding questionnaire and feedback on the application of OSCD in the graph used during the experiment.

4.7.1. PSSUQ Results

Figure 5 presents the PSSUQ scores for each question (Q1-19) and the metrics of the **expert** cohort.

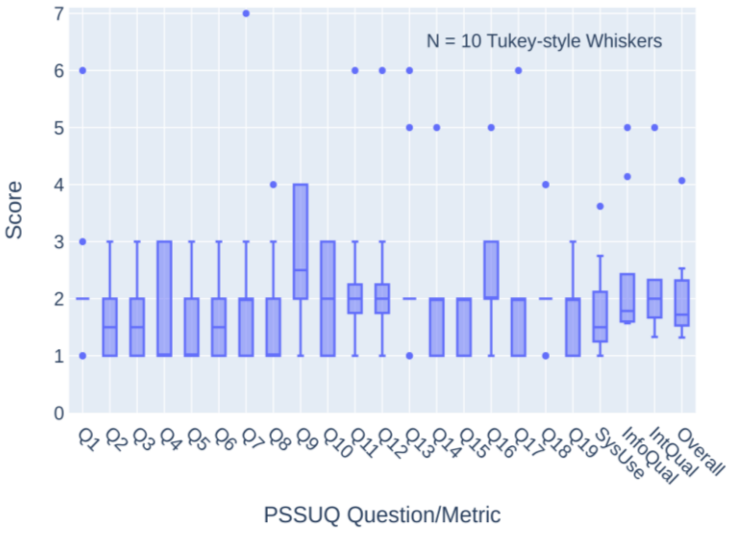


Figure 5: Box plot of PSSUQ scores for **expert** cohort

The metric scores were compared with acceptable thresholds found in research [10]. Each metric scored better than its threshold by at least 10%, which included system usefulness (56%), information quality (27%), interface quality (14%) and overall (38%). Most questions (18 out of 19) had a third quartile of 3 or less. Similar to the results of the student cohort, only one question (Q9) had a third quartile of 4, which related to error messages. The overall results of the PSSUQ indicate sufficient satisfaction with the metrics and questions scoring better than their respective thresholds.

4.7.2. Understanding Questionnaire Results

Table 5 presents the scores of the understanding questionnaire for the **expert** cohort.

Table 5: Summary of understanding questions results for **expert** cohort

Question #	Section 1 (μ)	Section 1 (σ')	Section 2 (μ)	Section 2 (σ')
Q1	1.00	0	0.70	0.46
Q2	0.90	0.3	0.80	0.4
Q3	0.80	0.4	0.95	0.15
Q4	0.90	0.3	0.90	0.3
Q5	1.00	0	1.00	0
Q6	1.00	0	1.00	0
Overall (μ)	0.93	0.17	0.89	0.22

Most (11 out of 12) questions have a mean score of at least 80% correct, which indicated an overall sufficient understanding by participants of the information presented to them by the MQI framework. In addition, the low standard deviation of both sections (< 0.25) indicated that the scores are clustered around the mean. However, the worst scoring question had (1 out of 12) 70% correct and related to a change description provided by the framework. This poor score could be as a result of tool-tips being incompatible with the browser of certain participants.

4.7.3. Results related to Application of OSCD

Each comment received from the **expert** cohort through the OSCD application questionnaire was reviewed to identify if a recommendation, by the expert, related to the ontology was indicated. Thereafter, it was considered by the lead researcher as to whether the recommendation should be addressed. An extract of recommendations received by experts and how they were addressed by the lead researcher is presented in **Table 6.**

Table 6: Extract of **expert** feedback on the application of OSCD

Recommendation	Method to Address
"Maybe by including the previous value for UpdateSourceData"	An Additional property "hasPreviousValue" was added to the ontology, which represents the previous value of the changed value
"provenance data related to who made the changes but it might be difficult to find that info in the ontology metadata. Also the time period the change has been made (after how long the change was made). But these are only minor things and only some suggestions to consider."	An Additional property named "wasChangedBy" was added to the ontology, which represents the agent who made the change.

The recommendations from experts resulted in the addition of two properties in OSCD. In addition, the other feedback affirmed sufficient applicability by providing comments such as *"It seems to be well presented."*, *"Seems clear to me"* and *"Seems like a useful tool"*.

4.8. Thematic Analysis

Thematic analysis was conducted in order to identify patterns in the qualitative data of both cohorts following the six-step process outlined in **Section 4.2**. The themes and codes[23] were created using a "bottom-up" approach, which involved defining them as they emerged from the data. The final report was produced using Taguette [14], which is a qualitative data tagging framework and presented the references for each code in the data. The themes and associated codes defined as a result of thematic analysis are presented in **Table 7**. The defined themes and codes were designed to group discovered negative and positive patterns. For instance, "Positive GUI Requirements" indicated patterns related to sufficient GUI requirements, such as aesthetic interface and clear layout, while "Negative GUI Requirements" indicated patterns related to insufficient GUI requirements. Therefore, the frequency of codes in the themes can be used to identify limitations of the usability of the MQI framework.

Table 7: Description and Occurrences of themes and related codes discovered in Thematic Analysis

Theme	Ratio	Description	Codes
User friendly	33.9%	The framework was easy to use and understand.	Easy to use, Efficient, Clear layout, Intuitive
Positive GUI requirements	19.4%	The layout and aesthetics of the framework are sufficient.	Aesthetic interface, Clear interface navigation, Clear layout
Positive user experience	16.5%	Positive user experience while interacting with the framework.	Straightforward, Error free, Adequate error recovery Quicker and easier to use over time
Negative GUI requirements	11.5%	The layout and aesthetics of the framework are not sufficient.	Unclear interface navigation, Unaesthetic interface, Unclear layout
Useful	9.35%	Functionality of the framework which was useful.	Overall usefulness, Drop-downs useful, Tool tips useful, Error messages useful
Clarify description and features	6.47%	Overly complicated and ambiguous text displayed on the framework.	Clarify text descriptions, Verbose, instructions, Ambiguous error message, Additional information required
Technical errors	2.88%	Technical errors which occurred during the completion of the experiment tasks.	Missing tool tip text description

The results of the thematic analysis indicated overall positive usability during the experiment with nearly 80% of codes related to positive themes, which included "User friendly", "Positive user experience", "Positive GUI Requirements" and "Useful". The most common negative codes were in the "Negative GUI requirements" theme and mainly related to the number of tabs which the framework opened during the experiment, which resulted in limited navigation.

4.9. Hypotheses

Based on the experiment undertaken, the hypotheses are examined below.

Hypothesis **H1**: The framework facilitates the identification of changes in source data and links with respective mappings. Based on an analysis of the experiment results gathered for both cohorts, it is reasonable to assert that Hypothesis **H1** is **supported**. The PSSUQ scores indicated that the usability provided by the framework was sufficient for completing the tasks with both scoring better than acceptable thresholds by at least 14%. The understanding questionnaire which provides evidence that the links between source data changes and respective mappings were understood scored high numbers in both sections for both cohorts. The results of section 1 for both cohorts scored an average of 87% correct. The results of section 2 for both cohorts scored an average of 89% correct. The average score of both sections in the questionnaire for both cohorts is 88% correct. The results indicated that participants with varying levels of knowledge were able to understand information related to changes in the source data of respective mappings. In addition, the most common themes (Positive user experience, Positive GUI requirements, User friendly) discovered by thematic analysis identified patterns related to positive overall usability.

[23] Code descriptions at https://drive.google.com/file/d/1G7pIyl2QxdhsaaL49iMQiB1F-SzHW_PS

Hypothesis **H2**: The participants' background knowledge influences the successful completion of the tasks. Based on an analysis of the experiment results gathered for both cohorts, it is reasonable to assert that Hypothesis **H2** is **not supported**. The satisfaction of the usability which was measured through the PSSUQ indicated that participants in both cohorts had similar levels of satisfaction. The scores of PSSUQ for both cohorts scored similarly better than acceptable thresholds found in research with a mean of 44% better for students and 34% for experts. Furthermore, the results of the understanding questionnaire indicated that participants in both cohorts similarly understood the information provided by the framework. The scores of the understanding questionnaire were similar with a difference of 6% between their mean scores. In addition, the small difference of 0.04 between the standard deviations indicated that the scores of both cohorts are clustered close to the mean. Moreover, no participants in both cohorts required assistance in order to complete the experiment. Therefore, it can be concluded that participants with limited knowledge of semantic web technologies can successfully interact with the framework to complete the tasks.

5. Related Work

A comparative study [22] has been conducted which discusses existing approaches to detect, propagate and describe changes in resources and interlinks of linked data datasets. The study compares the approaches based on requirements derived from community use cases, related to aspects such as discovery, granularity level, change modelling and notification mechanisms. The survey provided inspiration for the development of certain aspects of the MQI framework, such as the change monitoring and notification mechanism.

The most similar approach [24] proposes a framework for supporting alignment between relational databases and RDF views. The approach focuses on R2RML [3] mappings, which are designed to transform relational data. Changesets are computed by the framework and contain information used to detect differences between two versions of datasets. The changesets are automatically computed using mappings, which transform instance data from a relational database into a target ontology. The formalism has been described as a simpler language than R2RML. Unlike the MQI framework, the approach has been designed specifically for relational data and does not provide support for heterogenous formats and respective RML [4] mappings. However, the work provided insights into the requirements for the MQI framework.

DSNotify (DataSet Notify) [13] is an approach designed to detect changes in linked data datasets. The changes detected include create, remove, move, update of resources in the dataset. The framework detects changes using a monitoring component, which periodically executes a SPARQL query on the dataset and allows specific instance types to be targeted. A feature vector is created for each triple in the data retrieved from the query, which can be used later for detecting change events, by comparing these vectors. The triples in the datasets are modeled using the DSNotify EventSet vocabulary, which was created by the researchers specifically for the use case. The modelling of resource changes in a machine-readable format provided inspiration for the development of OSCD [20], which models source data changes instead of resources.

DELTA-LD [21] is an approach which detects and classifies changes in resources and interlinks between two versions of linked data datasets. The approach classifies resources that have both their IRI and representation changed. In addition, the approach aids in selecting the same resource in a different version of data which can be used to update a dataset. The approach proposes the DELTA-LD change model, which is used to represent detected changes and includes an ontology with two levels of granularity. The change model provided inspiration for the categorization of changes in OSCD.

sparqlPuSH [12] is a flexible approach designed to enable the real-time notification and broadcasting of changes in RDF stores. Notifications are sent in real-time to any RSS or Atom reader. SPARQL query results are delivered through PubSubHubbub (PuSH) protocol[24] when new RDF data is detected. The approach allows users to subscribe to a subset of the content in an RDF store. The users will receive a notification message each time content in the subset has changed. The objective is to provide a push-model where users do not have to identify new changes themselves. The approach provided useful

[24] https://github.com/pubsubhubbub/PubSubHubbub

background information for the MQI framework as it provides push notifications, however, related to source data changes. To the best of our knowledge, the MQI framework is the only approach which provides a notification mechanism for changes detected in source data used to generate linked data.

6. Conclusion

The component of the MQI framework, which was evaluated in this paper, demonstrated the ability to facilitate the timeliness propagation of source data changes into resulting linked data. Therefore, supporting the preservation of alignment between mappings and source data used to generate linked data, resulting in improved quality of metrics in the freshness dimension [20]. Furthermore, the information captured by the framework can provide indications of suitability for the application of consumers and improve trustworthiness by providing additional provenance [20]. Moreover, the evaluation approach followed by the framework could be applied to similar tools in order to validate them with respective end users. The usability testing of the framework provided a method for collaboration with participants who are domain experts (i.e. mapping experts) and early-stage mapping engineers (i.e. students), with a large sample size (55 participants), when compared with existing approaches [12,13,21,24]. The grouping of participants allowed diverse feedback to be gathered, which was compared in order to identify the level of background knowledge required to successfully interact with the framework. The results indicated that expert and non-expert mapping engineers could benefit from use of the change detection component. In addition, it is hoped the additional functionality added since the evaluation will be a step closer to autonomic maintenance of alignment, by allowing software agents to understand detected changes and automatically take appropriate actions in order to propagate them and prevent a decrease in data quality [20].

Future work includes the completion of the implementation of the new functionality discussed in **Section 3**, designed to provide suggestions to agents to aid in improving alignment between the mappings and data sources used to generate linked data datasets. An evaluation will be conducted on the new functionality in order to ensure the framework provides sufficient usability for respective end users. The evaluation will be structured similar to the one described in this paper, however, slightly different metrics will be used. Satisfaction will be measured similarly using the PSSUQ [10], however, understanding will not be measured. Instead, the level of alignment between a provided mapping and source data will be compared before and after the tasks have been completed, therefore, identifying whether an improvement has been made.

Acknowledgements

This research was conducted with the financial support of the SFI AI Centre for Research Training under Grant Agreement No. 18/CRT/6223 at the ADAPT SFI Research Centre (Grant No. 13/RC/2106_P2) at Trinity College Dublin.

References

[1] Haytham Assem, Lei Xu, Teodora Sandra Buda, and Declan O'Sullivan. 2016. Machine learning as a service for enabling Internet of Things and People. *Pers. Ubiquitous Comput.* 20, 6 (2016), 899–914. DOI:https://doi.org/10.1007/s00779-016-0963-3

[2] Mokrane Bouzeghoub. 2004. A framework for analysis of data freshness. In *Proceedings of the 2004 international workshop on Information quality in information systems*, 59–67.

[3] Souripriya Das, Seema Sundara, and Richard Cyganiak. 2012. R2RML: RDB to RDF Mapping Language. *W3C Recomm.* (2012). DOI:https://doi.org/10.1017/CBO9781107415324.004

[4] Anastasia Dimou, Miel Vander Sande, Pieter Colpaert, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. 2014. RML: A Generic Language for Integrated RDF Mappings of HeterogeneousData. In *Proceedings of the Workshop on Linked Data on the Web co-located withthe 23rd International World Wide Web Conference (WWW 2014), 2014.*

[5] Miguel Grinberg. 2018. *Flask web development: developing web applications with python*.

O’Reilly Media, Inc.

[6] Steve Harris, Andy Seaborne, and Eric Prud’hommeaux. 2013. SPARQL 1.1 query language. *W3C Recomm.* 21, 10 (2013), 778.

[7] Lalana Kagal. 2002. Rei: A policy language for the me-centric project. (2002). DOI:https://doi.org/10.13016/M2MG5B-HRA9

[8] Holger Knublauch and Dimitris Kontokostas. 2017. Shapes Constraint Language (SHACL), W3C Recommendation 20 July 2017. *URL https//www. w3. org/TR/shacl* (2017).

[9] D Krech. 2006. Rdflib: A python library for working with rdf. *Online https://github. com/RDFLib/rdflib* (2006).

[10] James R. Lewis. 2002. Psychometric Evaluation of the PSSUQ Using Data from Five Years of Usability Studies. *Int. J. Hum. Comput. Interact.* 14, 3–4, 463–488. DOI:https://doi.org/10.1080/10447318.2002.9669130

[11] Lorelli S Nowell, Jill M Norris, Deborah E White, and Nancy J Moules. 2017. Thematic analysis: Striving to meet the trustworthiness criteria. *Int. J. Qual. methods* (2017).

[12] Alexandre Passant and Pablo N Mendes. 2010. sparqlPuSH: Proactive Notification of Data Updates in RDF Stores Using PubSubHubbub. In *SFSW*.

[13] Niko Popitsch and Bernhard Haslhofer. 2011. DSNotify - A solution for event detection and link maintenance in dynamic datasets. *J. Web Semant.* 9, 3, 266–283. DOI:https://doi.org/10.1016/j.websem.2011.05.002

[14] Rémi Rampin and Vicky Rampin. 2021. Taguette: open-source qualitative data analysis. *J. Open Source Softw.* 6, 68 (2021), 3522.

[15] Alex Randles, Ademar Crotti Junior, and Declan O’Sullivan. 2020. A Framework for Assessing and Refining the Quality of R2RML mappings. In *Proceedings of the 22nd International Conference on Information Integration and Web-Based Applications & Services* (iiWAS2020). DOI:https://doi.org/10.1145/3428757.3429089

[16] Alex Randles, Ademar Crotti Junior, and Declan O’Sullivan. 2020. Towards a vocabulary for mapping quality assessment. In *15th International Workshop on Ontology Matching collocated with the 19th International Semantic Web Conference (ISWC 2020), 2020*, 241–242.

[17] Alex Randles, Ademar Crotti Junior, and Declan O’Sullivan. 2021. A Vocabulary for Describing Mapping Quality Assessment, Refinement and Validation. In *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*, 425–430. DOI:https://doi.org/10.1109/ICSC50631.2021.00076

[18] Alex Randles and Declan O’Sullivan. 2021. Assessing quality of R2RML mappings for OSi’s Linked Open Data portal. *4th Int. Work. Geospatial Linked Data ESWC 2021* (2021).

[19] Alex Randles and Declan O’Sullivan. 2022. Evaluating Quality Improvement techniques within the Linked Data Generation Process. In *18th International Conference on Semantics Systems (SEMANTiCS).*

[20] Alex Randles and Declan O’Sullivan. 2022. Modeling & Analyzing Changes within LD Source Data. In *8th Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW) co-located with the 21st International Semantic Web Conference (ISWC 2022).*

[21] Anuj Singh, Rob Brennan, and Declan O’Sullivan. 2018. DELTA-LD: A Change Detection Approach for Linked Datasets. In *4th Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW) co-located with the 15th Extended Semantic Web Conference (EWSC 2018).*

[22] Jürgen Umbrich, Boris Villazön-Terrazas, and Michael Hausenblas. 2010. Dataset dynamics compendium: a comparative study. In *Proceedings of the First International Conference on Consuming Linked Data-Volume 665*, 49–60.

[23] J ¨ Urgen Umbrich, Michael Hausenblas, Aidan Hogan, Axel Polleres, and Stefan Decker. *Towards Dataset Dynamics: Change Frequency of Linked Open Data Sources*. Retrieved from http://code.google.com/p/pubsubhubbub/

[24] Vânia Vidal, Narciso Arruda, Matheus Cruz, Marco Casanova, Carlos Brito, and Valéria Pequeno. 2017. Computing changesets for RDF views of relational data. In *Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW 2017)*, 43–58.