Improving Quality in the Publication of LD

Alex Randles\(^a,\)* and Declan O’Sullivan\(^a\)

\(^a\) ADAPT Centre for Digital Content, School of Computer Science and Statistics, Trinity College Dublin

Abstract. A significant proportion of Linked Data (LD) is created through mapping of data from a variety of sources of data. These mappings define the transformation rules from non-graph based data into graph based Resource Description Framework (RDF) data. The definition of mappings is a complex and time-consuming task which is prone to errors. Oftentimes, the resulting linked data datasets have varying levels of quality. In addition, quality issues are commonly detected after the mapping artefact has been executed and the linked data has already been published. Quality issues in mappings can result in an exponential growth of issues in the resulting dataset, thus greatly decreasing overall quality. In addition, 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. In addition, capturing information detailing mapping quality and source changes in a machine-readable format will allow software agents to automatically process them and take appropriate actions to preserve the quality of mappings used to create the linked data datasets. It is argued in this article that addressing quality issues within the mapping artefacts will positively improve the quality of the resulting dataset that is generated. Evaluating an approach designed to improve and maintain declarative uplift mappings involved in the publication of linked data is important to provide evidence of sufficient usability. In addition, evaluation allows the requirements of the approach to be validated with domain experts. This article describes the evaluation of the Mapping Quality Improvement (MQI) Framework which aims to guide linked data producers in producing high quality datasets, by enabling the quality assessment and subsequent improvement and maintenance of the mapping artefacts. The evaluation of the MQI framework and associated ontologies used diverse instruments and involved over 100 participants with varying levels of background knowledge.

Keywords: Linked Data; Publication; Quality Improvement; Evaluation.

1. Introduction

Linked data datasets are being published onto the web of data at an exponential rate with currently over 1200 datasets published on the Linked Open Data (LOD) cloud\(^1\). The LOD cloud provides a collaborative platform for publishing datasets according to LOD principles. The datasets contain knowledge about domains such as financial, medical, environmental, among others. The quality of the data will affect how useful data consumers find the data for their application. Data quality is often referred to as “fitness for use” [1,2] and is a multidimensional concept which is determined by the stakeholders and factors involved in the creation of the data [3]. Currently, quality assessment within the linked data domain is performed on published data and is often the responsibility of data consumers rather than the producers [3]. Research [5] has demonstrated the highly variable level of quality on the LOD cloud with an aggregated conformance of 60% of the datasets with respect to the 27 metrics which cover aspects such as accessibility, licensing, provenance, among others. Interestingly, metrics such as undefined terms (54%), incorrect domain/range (60%), licensing (11%) and basic provenance (12%) scored worse. More recent research [4,5] has identified that the current quality of the LOD cloud remains poor.

\(^*\) Alex Randles, ADAPT Centre for Digital Content, Trinity College Dublin, Dublin 2, Ireland; Email:alex.randles@adaptcentre.ie.

\(^1\) http://cas.lod-cloud.net/
Oftentimes, these datasets are created with “uplift” mappings which define transformation rules from heterogeneous source data to RDF [8,10]. RDB to RDF Mapping Language (R2RML) [6] is a prominent uplift mapping language designed to express customized rules for transforming relational data to RDF. RDF Mapping Language (RML) [7] is another prominent uplift mapping language, which extends the existing R2RML vocabulary to support source data formats, such as XML, JSON and CSV. Both of these languages are RDF-based. YARRRL [8] provides a human readable text-based representation of RML mappings, which is hoped to decrease workload. However, despite the various representations available, creating error-free mappings is a complex and time-consuming task [3,9]. In addition, the declarative mappings used to generate the dataset often contain the root cause of many quality issues present in dataset [3,9]. Thus, addressing quality issues early in the publication stage will prevent them from exponentially growing in the resulting linked data. In addition, it is far more efficient to improve the quality of mappings when compared to improving quality of large datasets due to the memory requirements involved [3,9]. Maintenance of mappings after execution when quality requirements are satisfied is essential to prevent a decrease in quality over time [10]. Linked data has been described as highly dynamic in nature due to continually changing data [11]. Oftentimes, the level of alignment between mappings and respective data sources can be impacted due these changes [10]. The decrease in alignment impacts the “freshness” of the dataset [10]. Freshness relates to the age and occurrences of changes in data and has been described as one of the most important aspects of linked data quality [12]. Thus, changes should be continuously identified and propagated into the mapping artefacts and resulting datasets to ensure that a highly accurate representation of the underlying data sources is provided to consumers [10].

This article describes an evaluation which was undertaken to validate an approach designed to improve the quality of mappings used in the linked data domain, in addition, to resolving identified state of the art limitations. The research question investigated in this article is “To what extent can the detection of declarative mapping quality issues and source data changes, facilitate the creation and maintenance of high-quality Linked Data (LD) datasets?”. The evaluation strategy contained 5 experiments using standardized evaluation instruments. The experiments consisted of usability experiments (2), an accuracy experiment (1) and expert validation experiments (2) which in total involved over 100 participants with varying background knowledge including knowledge engineer students, ontology development specialists and mapping specialists. In addition, 30 mappings collected from real world student projects and from independent research projects were used in the experiments. Usability testing, a method for collaboration between computer scientists and domain experts [13], was used to iteratively refine and validate the design of the developed MQI Framework [9,10,14–16] and two associated ontologies: Ontology for Source Change Detection (OSCD) [10,14] and Mapping Quality Improvement Ontology (MQIO) [17,18] were designed to facilitate the generation of high-quality mappings.

The remainder of this article is structured as follows: Section 2 provides an overview of relevant work related to addressing mapping quality and the dynamics of linked data. Section 3 describes the proposed approach designed to resolve the identified limitations. Section 4 discusses the evaluation approach applied in order to validate the MQI framework. Section 5 discusses lessons learned from conducting the research presented in this article. Section 6 presents conclusions and outlines next steps.

2. Related Work

This section discusses the review of related work which was conducted and resulted in the discovery of two key state of the art limitations (See Section 2.3). The reviewed approaches are grouped based on their association to the research objectives targeted in this work, which included “Approaches to support mapping quality assessment and refinement” and “Approaches to support mapping alignment”.

2.1. Mapping Quality Assessment & Refinement

The state of the art in mapping quality frameworks for linked data has been reviewed. Evaluating the quality of linked data tools with potential end users will demonstrate the usefulness of the design [6]. While several of the approaches in the state of the art have been adopted by users within the community, none of the approaches described have published an evaluation which studies user interaction. Most of these approaches have been evaluated using a system evaluation, while the evaluation described within this
article has used a large sample size of users and standardized usability methods.

EvaMap [19] is a mapping quality framework used to assess and improve the quality RDF mappings. The work uses YARRRML [8] mappings, which are a human readable representation of RDF mappings. The framework uses a set of metrics organized into 7 dimensions to assess the quality of the mappings or the resulting datasets when instances are required. Weights can be associated with metrics to provide different importance. Furthermore, a global quality score is generated to represent the overall quality of the mapping. Moreover, feedback is provided to users on how to improve the quality. The reports generated by the framework are human-readable and not machine-readable. An evaluation has not been completed on the framework from what we have seen published.

The approach presented in [2] extends an existing linked data quality assessment framework named Luzzu framework [20]. The approach focuses only on quality assessment and does not concern quality improvement. R2RML mappings [6] are targeted by the approach and assessed using metrics which are commonly used to assess dataset quality. Luzzu is extensible which allows the users to add additional metrics to the framework. Four metrics have been implemented by the authors which relate to the representational category [2] of data quality. Luzzu generates two machine-readable reports, however, the problem report is the focus of the work. An evaluation was completed on mappings from real world use cases. The results show the potential to identify quality issues in certain cases. The approach was found to be reasonably accurate at identify quality issues, however, there was certain cases where ontologies could not be retrieved and queried.

Resglass [3] provides a rule-driven methodology to detect inconsistencies within the rules used to generate linked data datasets. The approach ranks rules and ontology terms in order that should be inspected by an expert based on a score. Refinements are completed by an expert. Inconsistencies within the dataset are used to refine the rules and ontologies again. The work provides an implementation which targets RML [7] mappings. The inconsistencies are detected using a rule-based reasoning system [9]. The methodology has been applied to two real-life use cases DBpedia and Computer Science bibliograph (DBLP). The researchers discuss manual refinements which could potentially be used to remove these inconsistencies.

The approach [21] provides a test-driven approach for mapping assessment and semi-automatic refinements based on the quality assessment. The implementation targets RML [7] mappings and extends RDFUnit [22] which is an RDF test-case-based architecture. The RDFUnit test cases are extended to apply to mappings by adjusting the assessment queries. The semi-automatic refinements query the RDFUnit serializations of the quality information which enables triples to be add/delete or suggest actions to the user. The evaluation was applied to diverse use cases which included DBpedia and iLastic. The assessment of the mappings collected detected a large number of quality issues and a discussion of possible semi-automatic refinements. The results indicated that assessing mappings is more efficient in terms of computational complexity and requires significantly less time compared to assessing the dataset.

2.2. Dynamics of LD

Similar to section 2.1, none of the existing papers reviewed in this section have included any usability testing that may have been undertaken to validate them with respective end users.

A comparative study [11] 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 [23] to the MQI framework, proposes a framework for supporting alignment between relational databases and RDF views. The approach focuses on R2RML [6] 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 [7] mappings. However, the work provided insights into the requirements for the MQI framework.
DSNotify (DataSet Notify) [24] 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 [10], which models source data changes instead of resources.

DELTA-LD [25] 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 Internationalized Resource Identifier (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 [26] 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 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 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.

2.3. State of the Art Limitations

The work described in this article is aimed at resolving two key limitations identified in as a result of the state of the art review.

Limitation 1: Usage of Non-Domain Ontology for Captured Information. While approaches exist to support the improvement of mapping quality and maintenance. None of the approaches express information using an ontology designed specifically to represent mapping quality metadata and respective source data.

Limitation 2: Lack of User Testing. None of the reviewed approaches have published details of what (if any) formal user evaluation undertaken, and it is not clear the level of validation for respective end users.

Through our research, Limitation 1 has been addressed through the development of the ontologies MQIO (Section 3.3) and OSCD (Section 3.4), with both being ontologies designed to support the capture of mapping quality and source change data. Limitation 2 was addressed by undertaking a comprehensive evaluation of the MQI framework (Section 4).

2.4. Research Contributions

Four contributions are discussed in this article which were designed to resolve the stated limitations.

MQI Framework. A framework designed to support agents in the identification and removal of quality issues in declarative uplift mappings. In addition, the framework provides quality-oriented provenance in a domain specific ontology-based format. It is hoped the framework can be used to bring quality improvement procedures early into the publication process. The implementation of the framework facilitated the validation of the design with respective end users.

MQIO. An ontology designed to allow agents to represent mapping quality assessment, refinement and validation information in an ontology-based format. It is hoped the ontology can facilitate the creation of high-quality mappings, thus improving the quality of resulting datasets.

OSCD. An ontology designed to allow agents to represent source data changes in an ontology-based format. It is hoped the ontology can facilitate the maintenance of high-quality mappings, thus improving the level of alignment with source data.
Evaluation Strategy. The evaluation strategy applied to MQI framework includes multiple methods, metrics and participants which could be applied to validate similar or future quality assessment and improvement approaches.

Fig. 1. Overview of Mapping Quality Assessment and Refinement Component of the MQI Framework

3. Design and Implementation

3.1. MQI Framework

The MQI framework\(^2\) was designed to improve and maintain the quality of uplift mappings used in the publication of linked data. The objective of the framework is to support agents involved in publication by bringing quality assessment procedures early into the process. It is hoped the framework can be used to eradicate the root cause of many quality issues, which will prevent the exponential growth of issues into the resulting linked data dataset.

3.1.1. Mapping Quality Assessment & Refinement Component

The Mapping Quality Assessment and Refinement Component [9,15] (Figure 1) of the framework was designed to support agents in improving the quality of declarative uplift mappings used to generate linked data datasets. Initially, the component targeted R2RML [6] mappings, however, it was extended to support RML [7] mappings in order to improve mapping language coverage. First, an R2[RML] mapping is input into the framework. Local ontologies can also be input, which refer to those not published online. The framework will attempt to retrieve published ontologies which are reused in the mapping by fetching them online. A suite of quality metrics\(^3\) inspired from the metrics commonly used to assess dataset quality [2,27] are used to assess the quality of mappings. The metrics used by the framework are designed to assess the quality of mappings using three different aspects of artefact quality, which are mapping quality, data quality and vocabulary quality [28].

Mapping Quality Aspect (MP). Firstly, the quality of the mapping itself is considered, by assessing, for instance, the extent to which the mapping correctly conforms to the specification of the mapping language used. Such as whether there are any redundant definitions within the mapping (which would affect the performance of mapping engines used during execution).

Data Quality Aspect (D). The second aspect relates to the quality of the output generated by an engine processing the input data and the respective mapping. Poor design decisions made at the stage of defining a mapping artefact, such as using non-dereferenceable classes and properties, or incorrect semantics, will decrease the quality of the dataset. This aspect focuses

---

\(^2\) MQI Framework at https://mqi-framework.adaptcentre.ie/

\(^3\) Listing of Quality Metrics at https://drive.google.com/file/d/1vCxeCK08MuG4u/wSMnlyea6TPTkT-J
on the quality of the output data which can be identified and fixed during mapping design-time.

**Vocabulary Quality Aspect (VOC).** Finally, the third aspect considered relates to the quality of the vocabularies used within the mapping, by assessing for instance, that these classes and properties contain human readable labels or comments in the vocabulary. The rationale for including this aspect is to ensure the quality of the resulting datasets by making quality information related to the vocabularies being used in the mapping transparent to mapping engineers.

The semi-automatic refinements provided by the framework involve a human-in-the-loop and have been inspired by previous related research [3]. The refinements can be categorized into the following three categories.

**Insert custom value.** Inserting a custom value involves users entering an IRI or Literal value within a text box. Thereafter, the framework will replace the violation value within the mapping with the value entered. Prefixes are provided on the framework which could help to decrease workload when defining each IRI. For instance, if an undefined property is used within the mapping, users can select a prefix and enter the remaining IRI within the text box, which will replace the respective undefined property.

**Select from suggested values.** Selecting a value from suggested values involves the users browsing a drop-down menu and selecting a value. These values are designed to resolve the quality issues. Thereafter, the framework will replace the quality violation value with the selected value. For instance, if an undefined property is used within the mapping, defined properties within the same namespace will be suggested to the user. Thereafter, users will select one of the values and the framework will replace the respective undefined property.

**Insert suggested value.** Inserting a suggested value involves the framework suggesting only one value to the user, which could hopefully resolve the quality issue. If the users are satisfied with the suggestion, the value will replace the violation value. For instance, if a datatype defined within the mapping does not match the datatype defined within the respective ontology. The datatype from the ontology will be suggested to the user and when selected will replace the respective datatype.

Once the refinements have been executed, a validation bar chart is displayed which shows the relationship between each quality violation and their corresponding quality dimension [1] in linked data quality. The refined mapping generated by the refinements is available to download on the framework by pressing a button. The quality and refinement reports (Appendix A) generated by this component are represented in RDF format expressed in MQIO (See Section 3.3). The usability and effectiveness of this component was tested with 58 participants, as outlined in Section 4.4.

### 3.1.2. Source Change Detection Component

The framework includes a component (Figure 2) designed to detect and relate source data changes with

---

4 Listing of Quality Refinements at https://drive.google.com/file/d/1TC_eRDupHg84qcm17km9jGZ6xXn9YE
respective mappings, named the source change detection component.

Various types of changes can be detected by the component which could impact the level of alignment [14]. The types have been inspired from existing approaches to capture changes in resources [24–26], interlinks [24,25] and source data [23]. Insert/Delete changes represent the addition or removal of data from the source. Move changes represent the relocation of the source data. Merge changes represent source data joined with other data. Datatype changes represent a change in the interpretation of data. Update changes represent when a value is reassigned. The component provides the ability to define a notification policy which includes details for when agents will be notified of changes. For instance, an agent may want to be notified after 5 insert changes have occurred.

Changes are repeatedly detected, uplifted and analyzed to identify potential impact to the level of alignment between mappings and respective source data. In addition, certain mappings may need to be re-executed in order to preserve the freshness of the linked data [14]. The source change reports (Appendix B) generated by this component are represented in RDF format expressed in OSCD (See Section 3.4). The usability and understanding of this component were tested with 55 participants as outlined in Section 4.5.

3.2. Interaction of Framework Components

Relevant data in the reports generated by both components of the framework is linked using SPARQL queries. In addition, data from the respective mappings is linked to identify aspects potentially impacting quality. Figure 3 presents an overview of the interaction of both components in the MQI framework.

![Fig. 3. Interaction between the two components of the MQI Framework](image-url)

For instance, source data changes can be linked with mappings which have been assessed to identify deleted or updated data references. Thus, mappings can become incompatible with the current version of data which could be addressed by information contained in the change reports, such as similar data references.

The framework was implemented as a Python Flask web application containing a graphical user interface (Figure 4) to support user interaction. SPARQL Queries (Appendix C) [30] are used to link RDF data generated from input into both components presented, such as quality issues and relevant source changes.

![Fig. 4. Implementation of Mapping Quality Assessment and Refinement (left) and Source Change Detection (right) components](image-url)

---

5 Listing of Change Types at https://drive.google.com/file/d/1X-e2RS7PqWXMp1LjA9e8_BW_RBx_1CV-

6 Video demonstration at https://drive.google.com/file/d/14-AQloIL9WnQ1iUwqszYfscQaKo0u_IZ
3.3. MQIO

MQIO [18,31] was designed to represent and interchange information related to the quality assessment, refinement and validation of mappings. MQIO is used by the mapping quality assessment and refinement component of the framework to represent entities and activities involved in the quality improvement of uplift mappings. Thus, quality reports can be linked with respective mappings to provide indications of suitability to agents [9]. Figure 5 presents the concepts and relationships of MQIO.

Fig. 5. Class interaction diagram of the MQIO

**Mapping Quality Assessment.** In this stage, the mapping artefact (mqio:MappingArtefact) is assessed. The assessment activity is captured through the mqio:MappingAssessment class. The agent who initiated the process is also captured (prov:Agent). The PROV Ontology (PROV-O) [32] is a W3C recommendation for representing provenance in linked data which is reused and extended to represent certain aspects related to agents, activities and entities. A mapping assessment activity may have quality requirements which are captured through the mqio:QualityRequirement class. This information allows the ontology to validate whether such quality requirements have been satisfied in the assessment and refinement stage. The ontology draws inspiration from the Data Quality Vocabulary (DQV) [33] where data quality was classified into categories (dqv:Category), dimensions (dqv:Dimension) and metrics (dqv:Metric). DQV was designed to describe the quality of linked data datasets. The ontology uses the information in such classes to generate a mapping validation report (mqio:MappingValidationReport). Each violation identified is then represented with the mqio:MappingViolation class.

**Mapping Quality Refinement.** This stage captures mapping refinements executed on the mapping. Each metric described using the ontology may have multiple refinements (mqio:MappingRefinement) associated with it depending on the quality aspect being measured. The refinement executed in the mapping is associated with the identified violation through the property (mqio:wasRefinedBy). In addition, refinements have scores representing the likelihood the respective violation will be resolved.

**Mapping Quality Validation.** Finally, the ontology provides quality information on the original mapping being assessed and the mapping which has been generated as a result of the refinement. As mentioned, each mapping assessment process may have quality requirements which can be validated at this stage. For instance, one may define a quality requirement for understandability related quality metrics to be of a particular value. In this stage, this can be validated by comparing the defined quality requirement and the resulting one. Requirements can be validated by identifying quality measurements (dqv:QualityMeasurement) associated with the quality assessment of the mapping.

It was decided to not reuse DQV directly as it does not include suitable concepts and relationships for representing the involved entities and activities, such as quality assessment (mqio:MappingAssessment) and refinement (mqio:MappingRefinement). While DQV

---

7 MQIO Specification at https://www.w3id.org/MQIO/
reuses PROV-O for activities (prov:Activity), it does not provide suitable concepts to distinguish between the semantics of these activities. Thus, it was decided to extend PROV-O to represent domain specific activities and associated entities.

3.4. OSCD

OSCD\(^8\) [31] was designed to represent and interchange information related to the changes within the source data of mappings. OSCD is used by the source change detection component of the framework to represent changes in the source data of respective mappings, thus allowing relationships to be discovered through inference. Figure 6 presents the concepts and relationships of the OSCD.

![Class interaction diagram of the OSCD](image)

The detection of changes within the source data is initiated by an agent who is responsible for maintaining the data (oscd:hasMaintainer). Changes will be detected between a previous version (oscd:hasPreviousSource) and the current version (oscd:hasCurrentversion). Once the change detection process has been initiated, changes are detected between when the process began (oscd:hasDetectionStart) and a predetermined end date (oscd:hasDetectionEnd) which is defined by the maintainer. A notification policy (oscd:hasNotificationPolicy) can be defined which is represented with the Rei policy ontology [34]. The notification policy allows agents to be notified of changes when a certain amount has occurred. Thus, ensuring the maintainers are informed of changes when required.

Changes which occur in a specific source data are grouped into a log (oscd:ChangeLog). Each change (oscd:hasChange) within a change log is represented as a specialized event (lode:Event) [35] which has resulted in a change in the source data used to generate linked data. An ontology for Linking Open Descriptions of Events (LODE) [35] was designed to publish descriptions of events as linked data. The reuse of LODE draws inspiration from the DSNotify approach (Section 2.2) which extended the ontology to represent resource level changes. Different change types exist within the ontology which allow the cause of the change to be captured. For instance, a value is inserted (oscd:InsertSourceData) in the source data. Additional information related to the change such as the data inserted (oscd:hasChangedData) or its location within the source data (oscd:hasDataReference). The model can be extended to capture domain specific changes.

4. Overview of Evaluations

The evaluation described in this article consisted of 5 experiments. The experiments were designed to evaluate the approach proposed to address the state of the art limitations identified by the researchers. Experiment 1 was a system evaluation. Experiments 2-5 involved respective end users, including experts in the mapping and ontology development domain. Finally, the application study applied the framework to two real world use cases. The evaluation approach is further described in this section. Data related to the evaluations described in this article is stored in a GitHub repository\(^9\).

4.1. Evaluation Strategy

Figure 7 summarizes the evaluation strategy used to validate the MQI framework (comprising the two components and two ontologies).

---

\(^8\) OSCD Specification at https://www.w3id.org/OSCD/

\(^9\) GitHub repository containing additional data at https://github.com/alex-randles/Journal-Supplementary-Data/
In addition, the publication of each ontology captured expert feedback which was used to further refine the design.

Experiments 4 and 5 (See Section 4.6). The refined versions of each ontology were reviewed by experts who specialize in ontology development, which provided validation for the design of both. Finally, the design of the MQIO and OSCD were consolidated by conducting experiments in order to gather expert feedback on the development process of both ontologies.

Application Study (See Section 4.7). The design of the framework was consolidated by applying it to two real world use cases. The first use case involved applying the framework to a project conducted by Ordnance Survey Ireland (OSi)\(^\text{11}\). The second use case involved applying the framework to a project conducted by Ericsson Software Technology (EST)\(^\text{12}\). Both use cases involved the transformation of relevant data using uplift mappings into RDF format.

4.2. Experiment 1 – Accuracy Experiment

The first evaluation completed on the framework involved testing the accuracy when detecting quality issues in real world mappings. 30 R2RML mappings were collected from research projects (20 mappings) and student MSc projects (10 mappings). The data being transformed included personnel, environment, historical and medical data. The mappings were assessed using a suite of quality metrics designed to detect issues relating to the conformance to mapping specification and reuse of data and vocabularies.

The results indicated sufficient ability to detect quality issues as a total of 228 quality issues were detected in the mappings. The issues included undefined classes, undefined properties, incorrect datatypes, among others. Likely the issues would multiply into the resulting dataset, thus, greatly decreasing the quality of the data [3,37]. Table 1 presents an overview of the quality issues detected in the experiment. Quality issues detected by the framework were compared to a manual examination completed following a checklist\(^\text{13}\). Problems which resulted in incorrect detections were resolved and the respective mapping reassessed. The mean number of issues per mapping is 8.

---

\(^{10}\) https://blog.pics.io/11-best-file-comparison-software/

\(^{11}\) https://osi.ie/

\(^{12}\) https://www.est.tech/

\(^{13}\) Manual Examination Checklist at https://drive.google.com/file/d/1c--TuXoxGhjugvR-Xs2htNT0oTZwZ7B
The results indicated most authors have a good understanding of conformance to mapping specification (Mapping Quality) with 5 quality violations resulting from these metrics. Metrics related to data quality (Data Quality) and quality of the reused vocabularies (Vocabulary Quality) scored 111 and 112 quality issues, respectively. Thus, a similar level of conformance was observed for these aspects of quality. The most common quality issues in the data quality aspect related to terms undefined in their respective namespace. For instance, a large number of undefined terms (openfit:weight) were slightly misspelled (openfit:weigh). The most common quality issues in the vocabulary quality aspect related to the lack of licensing information in the reused vocabularies.

Students contributed 10 out of the 30 tested mappings, while accounting for 93 (41%) of quality issues which in comparison is more than research mappings. These results indicated the level of background knowledge of respective authors influenced the quality of mappings. Thus, it is important to validate the quality of mappings prior to execution and publication of the resulting dataset [3,9,21]. In addition, the quality-oriented can provide indications of suitability for the application of consumers [1,2].

4.2.1. Lessons learned from Experiment 1

The following lessons were learned as result of testing the accuracy of the mapping quality assessment and refinement component.

**Differences in Ontology Designs.** Differences in ontology designs must be considered when executing metrics which compare terms in the mapping to respective ontology terms. For instance, certain properties include a complex domain (owl:unionOf) such as prov:atLocation defined in the PROV-O [32]. While other properties in PROV-O do not include a complex domain. Thus, it is essential to use queries which consider multiple domains and a single domain when validating whether the correct domain is defined in a mapping. Similar considerations should be applied when validating the range and other aspects influencing mapping quality.

**Consideration for Generalized Concepts.** Generalized concepts such as rdfs:Class [38], owl:Class [39] and xsd:anyURI are should be considered when comparing mapping terms to ontology terms. For instance, when validating the usage of incorrect datatypes, xsd:anyURI will allow any XML Schema datatype to be defined. A direct comparison between the datatype and a datatype defined in a mapping such as xsd:string would result in an incorrect detection of a mapping quality issue. Similar, a domain or range defined rdfs:Class will allow any RDF resource to be used as all resources defined in RDF are inherently of this type. Thus, filters should be used when making a comparison to handle specific cases.

4.3. Metrics used during Usability Testing

The two usability experiments described in this article used the following metrics:

**Post-Study System Usability Questionnaire (PSSUQ).** The PSSUQ [13] is used to measure users perceived satisfaction of a software system. It was decided to use the PSSUQ rather than similar questionnaires such as the System Usability Scale (SUS) as extensive psychometric evaluation has been completed on the PSSUQ. In addition, the PSSUQ contains open-comment sections to capture qualitative data while SUS does not.

The questionnaire provides the ability to do standardized comparisons with other systems or evolutions of the system. The PSSUQ uses a 7-point Likert Scale where the lower score results in higher satisfaction. The second version of this questionnaire which includes 19 questions was used for the study. In addition, each question includes an open-comment section for providing additional feedback about different aspects of the usability. The PSSUQ includes

---

**Table 1: Quality issues detected in Experient 1**

<table>
<thead>
<tr>
<th>Metric ID</th>
<th>Violation Count</th>
<th>Metric Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping Quality</td>
<td>5</td>
<td>Language tag not defined in RFC 5646.</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>Usage of undefined class</td>
</tr>
<tr>
<td></td>
<td>57</td>
<td>Usage of undefined property</td>
</tr>
<tr>
<td></td>
<td>36</td>
<td>Usage of incorrect domain</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>Usage of range</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Usage of incorrect datatype</td>
</tr>
<tr>
<td>Data Quality</td>
<td>1</td>
<td>Human readable datatype</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>Domain and range definitions</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Basic Provenance</td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>Machine readable licensing</td>
</tr>
<tr>
<td></td>
<td>45</td>
<td>Human readable licensing</td>
</tr>
<tr>
<td>Vocabulary Quality</td>
<td>1</td>
<td>Information</td>
</tr>
</tbody>
</table>

---

14. https://www.w3.org/TR/xmlschema-2/#anyURI

The task sheet was designed to identify who would meet the inclusion/exclusion criteria. These participants were recruited individually through email invitation and completed the experiment to contribute to the research objectives. The participants from the student cohort were recruited from the MSc Knowledge and Data Engineering (CS7IS1) module at Trinity College Dublin. Each member of the class had the option to complete the experiment as a portfolio task for the module.

**Background.** The participants within the expert cohort are Semantic web researchers who are very knowledgeable with RDF and the R2RML mapping language. These participants have previous experience in creating and executing R2RML mappings. Participants from the student cohort have little knowledge of the theory of the R2RML mapping language. In addition, these participants have little experience with creating R2RML mappings, however, they have basic knowledge of semantic web technologies.

**Participants.** The expert cohort consisted of 10 participants after the inclusion/exclusion was applied. The student cohort consists of 59 students, which was reduced to 48 participants after the criteria was applied to the cohort.

**Completion of Experiment.** The participants in the expert cohort completed the experiment synchronously using zoom video conferencing platform while their think aloud statements were being recorded. The participants from the student cohort completed the experiment asynchronously by accessing the framework using provided login details. In addition, the cohort did not require the use of a video conferencing platform as the think-aloud protocol was not used because it would not be feasible to arrange a zoom meeting for each 48 students and to transcribe/analyze their think-aloud statements.

**Task Sheet.** The task sheet for participant’s contained several tasks designed to test the main functionality of the component (quality improvement of mappings). Participants were asked to upload the provided mapping and review the quality assessment information. Thereafter, they were asked to select and execute the most appropriate refinement on the mapping and review the quality reports generated in MQIO. Finally, the last task was to complete the PSSUQ.
4.4.2. Evaluation Metrics

The following metrics were used during this experiment, in addition, to the metrics outlined in Section 4.3.

**Time Per Task.** The time per task can be used as a comparative measure to determine if certain factors such as a worse PSSUQ score have a relationship with their timing.

**Violation Count.** The violation count refers to the number of quality issues which have been resolved by the participant during the experiment. Three violations inspired by issues detected in experiment 1 were introduced in the mapping provided to participants. The number of violations within the refined mapping generated was used to determine how effective the framework is at improving the quality of mappings.

**Think-aloud protocol** [42]. The protocol was used to collect think-aloud statements, where participants verbalize their thoughts while completing the tasks. Only think-aloud statements were collected from the expert cohort as it would not be feasible to collect think-aloud statements from each of the 48 participants in the student cohort. The transcripts were initially generated by the conferencing platform and manually enhanced.

4.4.3. Results of Student Cohort

**Table 2** presents a summary of quantitative data results for the **student cohort.** The time for completion, PSSUQ metric mean scores and violation count within refined mapping were calculated.

<table>
<thead>
<tr>
<th>Time</th>
<th>Mean time</th>
<th>Median time</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSSUQ</td>
<td>10.06 minutes</td>
<td>9 minutes</td>
</tr>
<tr>
<td>Violation Count</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0 violations (Best case)</td>
<td>24 participants (50%)</td>
<td></td>
</tr>
<tr>
<td>1 violation</td>
<td>10 participants (21%)</td>
<td></td>
</tr>
<tr>
<td>2 violations</td>
<td>5 participants (10%)</td>
<td></td>
</tr>
<tr>
<td>3 violations (Worst case)</td>
<td>9 participants (19%)</td>
<td></td>
</tr>
</tbody>
</table>

The analysis starts by discussing the PSSUQ results, followed by the other quantitative data. The qualitative data is discussed in parallel with the quantitative data. The PSSUQ scores have been compared against acceptable thresholds within a previous research study [13] as no previous scores exist for this component of the framework.

**PSSUQ Results.** SysUse (+19%), InfoQual (+24%) and Overall (+16%) scored better than respective thresholds by at least 15%, thus indicating sufficient satisfaction for these aspects. However, the IntQual metric (-11%) scored worse than the respective threshold (2.49). These results indicated that participants were not satisfied with the interface quality which was supported by the frequency of codes in “Negative GUI Requirements” theme (Figure 8).

**Violation Count.** The original mapping contained 3 violations. 50% of participants have 0 violations. 71% have 1 or 0 violations. 29% have 2 or 3 violations. These results indicated that several students struggled to remove quality issues from the mapping, which could be influenced by the level of relevant background knowledge gained during the module.

**Timing.** The mean time for the student cohort is 10.06 minutes. The maximum time is 23 minutes and the minimum time is 2 minutes. The minimum time of 2 minutes based on the experience of the researcher could indicate certain students were not careful when completing the experiment. The fastest tasks related to the assessment process. The slowest tasks related to the selection/executing of refinements and the examination of the patterns within the validation report. These results indicated that the participants struggled to select refinements and interpret the validation report. Additional information could improve the time taken to select and execute refinements. The patterns within the validation report could be simplified to allow the participants to interpret the report more easily.

4.4.4. Results of Expert Cohort

**Table 3** presents a summary of quantitative data results for the **expert cohort.**

<table>
<thead>
<tr>
<th>Time</th>
<th>Mean time</th>
<th>Median time</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSSUQ</td>
<td>15.4 minutes</td>
<td>12.5 minutes</td>
</tr>
<tr>
<td>Violation Count</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0 violations (Best case)</td>
<td>9 participants (90%)</td>
<td></td>
</tr>
<tr>
<td>1 violation</td>
<td>1 participant (10%)</td>
<td></td>
</tr>
<tr>
<td>2 violations</td>
<td>0 participants</td>
<td></td>
</tr>
<tr>
<td>3 violations (Worst case)</td>
<td>0 participants</td>
<td></td>
</tr>
</tbody>
</table>

The analysis starts by discussing the PSSUQ results, followed by the other quantitative data. The qualitative data is discussed in parallel with the quantitative data. **PSSUQ Results.** SysUse (+65%), InfoQual (+24%) and Overall (+33%) scored better than respective thresholds by at least 20%, thus indicating sufficient satisfaction for these aspects. However, the IntQual metric (-9%) scored worse than the respective threshold (2.49). These results indicated that
participants were not satisfied with the interface quality which was supported by the frequency of codes in the “Negative GUI Requirements” theme (Figure 8).

Violation Count. 90% of participants have 0 violations in the refined mapping, while 10% have 1 violation in the refined mapping. No participants have 3 violations in the refined mapping. The low violation count within the refined mapping indicates that the framework could be an effective tool for helping an expert user to identify and remove quality violations.

Timing. The mean time for completing the experiment is 15.4 minutes with the fastest time being 11.05 minutes and the slowest time being 24.05 minutes. These results could indicate that not all experts could use the framework equally. In addition, noted during the experiment that some experts spent more time exploring the framework while others spent less time. The fastest tasks to complete were related to the assessment process. The participants took longer to choose and execute refinements. In addition, the slowest task related to examination of the patterns within the validation report. These results could indicate that the information provided relating to refinements could be improved to enable participants to select a refinement more easily. In addition, the layout of the validation report should be improved in future versions to improve the time it takes for participants to interpret the report.

4.4.5. Thematic Analysis

The six-steps (Section 4.4.5) of thematic analysis were completed on the qualitative data collected from the experiment, which included the open-comments of the PSSUQ and think-aloud statements for the expert cohort. Figure 8 presents the occurrences of themes which group codes18 defined as a result of the analysis.

User friendly (23.9%), Positive user experience (15.9%) and Useful (15%) themes accounted for nearly 60% of the codes. Thus, indicating that participants found the overall experience provided by the framework positive with the most common codes in these themes being “Easy to use” and “Overall Usefulness”. Comments such as “System was good!” and “very intuitive” supported these findings.

Negative GUI Requirements (19.5%) and Positive GUI Requirements (7.08%) themes indicated that more participants found the interface insufficient. Most code references related to the aesthetics of the framework rather than the layout and navigation.

Technical Errors (0.885%) related to blocked pop ups for certain participants which initially prevented graph downloads from the framework. However, most participants did not encounter this problem. Comments such as “So that was obviously blocked by my browser as a pop up” related to the error. The problem was resolved by altering the configuration of the security certification for the framework.

4.4.6. Comparison of Cohort Results

PSSUQ Scores. The perceived satisfaction for the mapping quality assessment and refinement component was sufficient for the SysUse, InfoQual and Overall metrics, with all scoring at least 10% better than their respective threshold for both cohorts. However, IntQual scored worse than the respective threshold for both the student (2.8) and expert (2.75) cohorts. Thus, the results indicated insufficient interface quality which requires improvements to improve the user satisfaction of both cohorts. As a result of these findings, it was decided to change the overall style of the graphical user interface (See Figure 4).

Violation Count. 90% of the expert cohort have 0 violations, while 71% of the student cohort have 0 or 1 violations in the refined mapping, which could indicate that the background knowledge of the expert cohort helped them to identify and remove the quality violations. In addition, no expert has 3 violations, while 10% of the student cohort had 3 violations. These results indicate that the effectiveness of the framework is influenced by the background

---

18 Listing of thematic themes and codes at https://drive.google.com/file/d/1G7plyl2QxdhssaAL49lMQb1F-SzHW_PS
knowledge. However, improvements previously mentioned could help students to identify and remove quality issues more easily.

**Timing.** The mean time for the expert cohort to complete the experiment is 15.4 minutes while the mean time for the student cohort is 10.06, which is about 5 minutes faster. The student and expert cohort have a median time of 13 and 12 minutes, respectively, which is only a difference of 1 minute. The majority of participants (Q3) completed the experiment in 20 minutes or less. However, the main difference is the maximum value. The student and expert cohort have a maximum time of 23 and 24 minutes, respectively, which could indicate that background knowledge does not influence the time taken to interact with the framework. The comparison of times between the cohorts is limited as the use of the think aloud protocol for the experts could have influenced completion time.

4.4.7. Lessons learned from Experiment 2

The following lessons were learned as result of testing the usability and effectiveness of the mapping quality assessment and refinement component.

**Verbal Statements were useful.** While transcriptions and examination of the verbal statements of the experts was time consuming, it helped to identify important aspects not captured by the quantitative metrics. Comments made by participants provided a mechanism to identify causes for scores of relevant quantitative metrics and the root cause of poor scores, which were used guide improvements. Thus, qualitative data significantly supported the refinement of the approach.

**Varying background knowledge.** The difference in background knowledge of the cohorts allowed the required level to be determined. It proved beneficial by identifying students required additional information to guide them through the identification of an appropriate refinements and associated values. As agents with varying levels of background knowledge publish linked data, identifying these limitations helps to improve usability for a larger potential audience.

**Minimal Background knowledge required.** Several refined mappings from students contained violations such as including data type names, such as `admingeo:a` or `date:xsd`, which are not data types. Other examples of violations included a property named `aair:http://www.w3.org/r2rml#`, which is undefined. These results indicated that a basic understanding of RDF technologies is required to successfully interact with frameworks, such as the MQI framework which was not designed to teach users relevant information.

4.5. Experiment 3 - Usability Testing

Experiment 3 [10] involved usability testing on the source change detection component of the MQI framework, whereas the first usability testing focused on the mapping quality assessment and refinement component. The evaluation was structured similar to experiment 2, however, the questionnaire was extended to test the understanding of the relationships detected by the new component.

4.5.1. Evaluation cohorts

The participants were grouped into two cohorts, which included an expert and student cohort. The cohorts’ recruitment process and background knowledge differed. Grouping participants into two cohorts allowed them to be characterized based on background knowledge.

**Recruitment.** Similar to experiment 2 (See Section 4.4.1) a student and expert cohort was used, however, this experiment contained 45 students and 10 experts.

**Background.** The participants in the expert and student cohorts had identical background knowledge to experiment 2 (See Section 4.4.1).

**Participants.** The expert cohort consisted of 10 participants after the inclusion/exclusion was applied. The student cohort consists of 50 students, which was reduced to 45 participants after the inclusion/exclusion criteria was applied to the cohorts.

4.5.2. Evaluation Structure

The structure of the evaluation was identical for both cohorts, other than the inclusion of one additional questionnaire section for the expert cohort. The additional section allowed experts to provide feedback on the design and application of OSCD [31]. The section was not included for the student cohort as their knowledge related to ontology design is limited. The structure of the experiment can be summarized as follows:

**Completion of Experiment.** Each cohort completed the experiment asynchronously and no conference call was required. Assistance was provided through email if required. The link to the questionnaire was provided on the framework after the participants had completed the tasks.
Task Sheet. The task sheet\textsuperscript{19} for participant’s contained several tasks designed to test the main functionality of the component (change detection of source data). Participants were asked to upload the details of the provided source data and review the displayed information. Thereafter, they were asked to identify links between specific changes and respective mappings. For instance, if a data reference has been renamed in the source data and not updated in the mapping. Only experts were asked to examine the graph expressed in OSCD, which documents the changes detected by the framework in the specific source data and other related information as shown in the specification [31]. Finally, the last task was to complete each section of the questionnaire.

4.5.3. Evaluation Metrics

The questionnaire\textsuperscript{20} used in this usability experiment contained four sections as outlined below:

Understanding Section. The first and second section within the questionnaire asked questions which related to information displayed on the framework. The information included the number of changes which have occurred, and which data references defined in the mapping no longer exist in the source data.

Ontology Application Section. The third section within the questionnaire for the expert cohort asked for feedback on the design and application of OSCD. The graph contained the source data changes detected by the application. The questions asked whether the ontology should include additional concepts/relationships (Q1) and suggested changes to the representation in the graph (Q2). In addition, an open comment section was included for additional feedback (Q3). It was decided to only ask experts as the students background knowledge of ontology design is limited.

PSSUQ Section. The fourth section within the questionnaire contained the PSSUQ (See Section 4.3) to measure user satisfaction.

4.5.4. Student Experiment Results

Figure 9 presents box plots representing the scores of the PSSUQ for the student cohort.

PSSUQ Results. SysUse (+63%), InfoQual (+46%), IntQual (+21%) and Overall (+47%) all scored better than respective thresholds by at least 20%, thus indicating overall sufficient satisfaction for the measured aspects. Comments such as “really satisfied” and “Simple yet effective” supported these findings.

Table 4 presents the mean score (\(\mu\)) and standard deviation (\(\sigma'\)) for each section (S) of the understanding questions for the student cohort.

Understanding Results. The overall mean score for both sections was greater than 80%. Thus, indicating the participants in the student cohort could successfully identify the total number of source data changes (S1.Q1), notification thresholds defined for the changes (S1.Q4, S1.Q5), the values changed (S2.Q3, S2.Q5) and data references in the mapping which have been changed (S2.Q2, S2.Q4, S2.Q6). The low standard deviations of the sections showed that the scores are not significantly dispersed, which indicates similar understanding by most of the participants.

\textsuperscript{19} Task Sheet at https://drive.google.com/file/d/1yyazk26j2My2ekdmwaiRZ_WRcYQXbh-qS

\textsuperscript{20} Questionnaire used during experiment at https://forms.gle/FMzH9fmFcy9yi5AH6
Thus, the overall results indicated a sufficient method to detect relationships which can be used to aid alignment between mappings and the underlying data sources.

4.5.5. Expert Experiment Results

Figure 10 presents box plots representing the scores of the PSSUQ for the expert cohort.

![Box plot of PSSUQ scores for expert cohort](image.png)

**PSSUQ Results.** SysUse (+56%), InfoQual (+27%), IntQual (+14%) and Overall (+38%) all scored better than respective thresholds by at least 10%, thus indicating overall sufficient satisfaction for the measured aspects. Comments such as “Easy to follow” and “So easy, very intuitive!” supported these findings.

Table 5 presents the mean score (μ) and standard deviation (σ’) for each section (S) of the understanding questions for the expert cohort.

<table>
<thead>
<tr>
<th>#</th>
<th>S.1 (μ)</th>
<th>S.1 (σ’)</th>
<th>S.2 (μ)</th>
<th>S.2 (σ’)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>1.00</td>
<td>0</td>
<td>0.70</td>
<td>0.46</td>
</tr>
<tr>
<td>Q2</td>
<td>0.90</td>
<td>0.3</td>
<td>0.80</td>
<td>0.4</td>
</tr>
<tr>
<td>Q3</td>
<td>0.80</td>
<td>0.4</td>
<td>0.95</td>
<td>0.15</td>
</tr>
<tr>
<td>Q4</td>
<td>0.90</td>
<td>0.3</td>
<td>0.90</td>
<td>0.3</td>
</tr>
<tr>
<td>Q5</td>
<td>1.00</td>
<td>0</td>
<td>1.00</td>
<td>0</td>
</tr>
<tr>
<td>Q6</td>
<td>1.00</td>
<td>0</td>
<td>1.00</td>
<td>0</td>
</tr>
<tr>
<td>Overall (μ)</td>
<td>0.93</td>
<td>0.17</td>
<td>0.89</td>
<td>0.22</td>
</tr>
</tbody>
</table>

**Understanding Results.** The overall mean score for both sections was greater than 90%. Thus, indicating the participants in the student cohort could identify the total number of source data changes (S1.Q1), notification thresholds defined for the changes (S1.Q4, S1.Q5), the values changed (S2.Q3, S2.Q5) and data references in the mapping which have been changed (S2.Q2, S2.Q4, S2.Q6). The low standard deviations of the questions show that the scores are not significantly dispersed, which indicates similar understanding by most of the participants. Thus, the overall results indicated a sufficient method to detect relationships which can be used to aid alignment between mappings and the underlying data sources.

4.5.6. Recommendations on OSCD

The expert cohort was asked to provide feedback on the application and design of OSCD, in addition to the concepts and documentation. Most of the feedback gathered did not contain recommendations on improvements and supported the current design, such as “Seems clear to me” and “It seems to be well presented.”. The recommendations received resulted in a number of minor changes to the OSCD, which included adding two properties to represent the previous value of a change (oscd:hasPreviousValue) and the agent responsible for the change (oscd:wasChangedBy).

4.5.7. Thematic analysis

The six-steps (Section 4.4.5) of thematic analysis were completed on the qualitative data collected from the experiment. Figure 11 presents the occurrences of themes which group codes defined as a result of the analysis.

![Pie Chart of themes discovered as a result of thematic analysis for Experiment 3](image.png)

---

21 Feedback on OSCD application at [https://drive.google.com/file/d/1AYzCSIKNCLnmu2tFiEB7sAAbX1gylfOx2](https://drive.google.com/file/d/1AYzCSIKNCLnmu2tFiEB7sAAbX1gylfOx2)

22 Listing of thematic themes and codes at [https://drive.google.com/file/d/1G7plyl2Qxhlssal49M0qB1FSzHW_PS](https://drive.google.com/file/d/1G7plyl2Qxhlssal49M0qB1FSzHW_PS)
User Friendly (33.8%), Positive user experience (16.5%), Useful (9.35%) and Positive GUI Requirements (19.4%): These themes accounted for nearly 80% of the codes. Thus, the analysis indicated participants found the overall experience provided by the framework positive with the most common codes in these themes being “Easy to use” and “Straightforward”. Comments such as “Very Good and clear” and “its user intuitive” supported these findings.

Positive GUI Requirements (19.4%) and Negative GUI Requirements (11.5%): The analysis indicated that more participants found the GUI sufficient. However, the occurrences of codes related to a negative GUI experience indicated that the interface somewhat needs improvement. Most comments related to negative GUI requirements identified the number of tabs as an issue such as “new tabs made it a little cluttered”. Technical Errors (2.88%) occurred infrequently, which indicated most participants did not encounter any. The main errors related to the missing tool tip text, which a small number of participants encountered. Comments such as “As mentioned, the info buttons didn’t work for me” related to the error. However, comments such as “There are help tool tips which make life easier,” indicated that the error was specific to certain machines.

4.5.9. Lessons learned from Experiment 3

Questionnaires can be Extended. Unlike experiment 2, the questionnaire used during the experiment was extended. Instead, the PSSUQ was added as section, which allowed additional feedback to be captured. The additional sections collected various quantitative and qualitative data which were useful during the analysis in order to collect evidence to support our findings. Reuse of a standardized provided a method to compare results with similar approaches in the state of the art.

Cross Platform Validation. The tool tips on the framework were compatible with most browsers, however, it was later identified that there not compatible with Safari web browser. Thus, it is important to validate the web application on all available browsers. The problem has since been resolved by changing the source code of the framework.

Limiting Tabs. Initially, the pages which detailed further information on aspects such as notifications and links with respective mappings opened in a new tab. It was identified that certain users struggled to navigate these tabs when referring back to the questionnaire. Thus, the pages will be limited to a single tab which is hoped to improve navigation.

4.5.8. Comparison of Cohort Results

PSSUQ Scores. The perceived satisfaction of the source change detection component was sufficient when all metrics were compared to acceptable thresholds found in research [13] with both cohorts scoring better. Similar levels of satisfaction were observed for the mean score better than the thresholds for the students (+44%) and experts (+34%). Thus, the results indicated that both cohorts were similarly satisfied with the user interaction of the framework.

Understanding Scores. The combined mean for both sections of the understanding questionnaire (88%) and low standard deviation (0.3) indicated sufficient understanding of relevant information provided by the framework. The low difference between the mean (6%) and standard deviation (0.06) of the cohorts scores indicated similar understanding.

4.6. Experiments 4 and 5 – Ontology Design Validation

Experiments 4 and 5 were conducted in order to validate the design of MQIO and OSCD, which involved 5 ontology design experts, who were not involved in the MQI development, each with more than 10 years of experience in ontology development. These participants were provided with documents23 detailing the design and evaluation methods applied during the development of each ontology. Thereafter, they were asked to provide feedback through a questionnaire24 on aspects within the respective document. Figure 12 presents an overview of the experimental structure.

23 Experiment Documents at https://drive.google.com/drive/folders/1uYmTMKde5UeGkZoHc
mUCh13SgMHn8pu

24 Ontology Design Questionnaire at https://forms.gle/LgV3ThJ4MrTcjkc7
The results collected from the 5 experts are summarized as follows.

**State of the Art.** All experts stated that both ontologies followed best practices in ontology design practices as recommended in the state of the art. Comments such as "It follows well established methods for developing and iteratively improving the concepts based on application." and "Use of both methodologies and state of the art ontology validation tools" supported these findings.

**Design Methodology.** 4 out of the 5 experts stated that design methodologies followed by both ontologies were sufficient. 1 expert stated that methodologies should include an additional two assessments. These assessments include conformance to the FAIR data principles [147], which were published in 2016 and proposed by a consortium of scientists and organizations. These principles are designed to guide data publishers in supporting the reusability of published data assets. The Grubers design principles (proposed in 1995) [59] provide comprehensive objective criteria’s designed to guide the development of ontologies. The five principles are 1) Clarity 2) Coherence 3) Extendibility 4) Minimal encoding bias and 5) Minimal ontological commitment, which encapsulate key criteria’s an ontology should satisfy. Both ontologies showed sufficient conformance to these assessments\(^{25}\).

**Concepts & Relationships.** 2 out of 5 experts did not provide recommendations for this aspect. The other experts suggested changing the type of a small number of properties from datatype properties (owl:DatatypeProperty) to object properties (owl:ObjectProperty) to capture additional information.

**Documentation.** One expert provided no recommendations to the documentation, while the other experts recommended the following: 1) include sample SHACL [43] constraints designed to assess the quality of graphs expressed in the respective ontologies and include and 2) add an appendix section with sample graphs rather than including them solely as hyperlinks. Thus, it was decided to add sample constraints for validating the quality of respective graphs and an appendix section detailing sample graphs to the documentation of both ontologies.

**Open-Comments.** 4 out of the 5 experts provided no recommendations through the open-comment section, with 1 stating “well done”. However, 1 expert recommended replacing the concept dul:Agent [44] with foaf:Agent [45] as it is more prominent for representing agents in the semantic web.

An iterative approach was adopted in order to assess and address each recommendation made by an expert. The process involved changing concepts and relationships using Protégé [46] ontology development tool. Protégé includes semantic reasoners which were used to ensure no inconsistencies were introduced by the changes to the ontologies. In addition, the updated ontology was assessed using OOPS! Pitfall Scanner [47] to detect common design pitfalls. Finally, the online documentation was updated and regenerated using Wizard for DOCumenting Ontologies (WIDOCO) [48] and republished on the web. Thus, it can be inferred that the final versions of the ontologies have sufficient quality in the view of these experts by addressing each of their recommendations.

### 4.6.1. Lessons learned from Experiments 4 and 5

The following lessons were learned as result of validating the design and evaluation methods of OSCD and MQIO.

**Further Refinement was Required.** Multiple sources of data, such as domain experts, publications and ontology evaluation tools were iteratively used to inform the refinement of the designs of both ontologies prior to the validation experiments. However, despite addressing the feedback, further refinement was required after the experiments. Thus, the additional feedback from ontology design experts helped to finalize the refinement and validation of the resulting

---

\(^{25}\) Conformance to Ontology Design Assessments at https://drive.google.com/file/d/1pKjtUFd2OHDj5xpwgeIAyfH7WsY
designs rather than relying on domain expert feedback alone.

**Diverse Views.** A sample size of 5 experts allowed varied feedback to be captured which was used to refine and validate both ontologies. The experts provided similar and contradictory responses which allowed us to compare and contrast the results to identify key required improvements. In addition, the agreement on key aspects, such as conformance to design practices in the state of the art provided solid evidence that a sufficient level of conformance was present.

**Prominent concepts are preferred.** Rather than reusing the same agent concept (dul:Agent) as LODE, which was extended in OSCD. However, experts recommended including a more prominent concept such as foaf:Agent. Thus, it is important to reuse the most prominent suitable concepts in the respective ontology to increase uptake.

### 4.7. Application Study

The MQI framework was applied to two real world use cases, which demonstrated the feasibility in real world settings. In addition, the approach was applied to publication processes involving two diverse knowledge domains. Thus, allowing the discovery and addressing of limitations when applied to different domains. The approach was used during the use cases in order to maintain high quality and fresh mappings. **Figure 13** presents an overview of interaction between the components of the framework with the environments of both involved use cases.

The **OSi use case** [15] involved the quality assessment and refinement of R2RML uplift mappings designed to transform Ireland's national geospatial data stored in the Prime 2 database [49]. The **source change detection component** was not applied for this use case as it had not been developed at this stage. The mappings reused ontologies, such as RDF Schema (RDFS) [38] and Dataset Quality Ontology (daQ) [50] to represent geospatial readings. Several quality issues were identified and removed during the use case, such as incorrect domain/range and undefined terms. For instance, the daq:MetricProfile concept is undefined in daQ, however, it was defined in a mapping. A semi-automatic refinement of the framework was used to identify similar concepts in the daQ namespace and it was later replaced with the daq:Metric concept which is defined. It was estimated based on manual examination that resolution of these quality issues potentially positively impacting 1750 triples.

The **EST use case** [16] involved application of both components of the framework. The use case involved inputting the R2RML mappings [6] and source data used by the Prometheus RDF Generator [51] into the respective components of the MQI framework. It was designed to transform time series metric data from a cloud native monitoring system named Prometheus into RDF format. The mappings were expressed in the Intent Based Control Loop Ontology (IBCLO) which contains concepts and relationships to represent metric information and associated intents/goals. The mapping quality assessment and refinement resulted in the identification and removal of several quality issues. The application of the source change detection component allowed changes to be identified and assessed. The notification policy was triggered several times which allowed changes such as new endpoints and server to be captured in timely manner.

#### 4.7.1. Lessons learned from Application Study

The following key findings were discovered as a result of applying the framework in the two use cases.

---

https://prometheus.io/
Auxiliary Usage. A secondary usage was identified for the mapping quality assessment and refinement component of the framework. The usage related to detection of inconsistencies in ontologies which are developed in parallel with mappings which reuse them. For instance, a quality issue detected related to an incorrect range being defined for a property. The range in the mapping was defined as datatype property (owl:DatatypeProperty), however, the ontology defined it as an object property (owl:ObjectProperty). It was later identified that the ontology under development contained an inconsistency which was resolved, and the respective mapping quality issue addressed.

Various domains. Applying the approach to use cases involving two different domains helped to identify limitations when improving mappings which transform diverse knowledge. In addition, the use cases involved mappings which reused diverse ontologies which provided evidence that MQI is not restricted to specific knowledge domains. Thus, the results of the application study indicated the MQI approach is applicable in different domains which is hoped to improve uptake.

Detection of Changes is Performance Intensive. It was emphasized through the application of the source change detection component that detecting and analyzing changes in large data sources was performance intensive. Thus, the inclusion of threading [52] in the implementation which allows detection and linking of relevant changes to be executed in parallel rather than sequential, which was hoped to improve overall execution time for users.

5. Discussion

This section presents the key findings which were discovered as a result of conducting the research discussed in this article.

5.1. Reuse of Existing Approaches

Previously, the authors were involved in creating a mapping quality assessment and refinement framework [28], which used SHACL constraints to execute quality metrics on mappings represented in RDF format, such as R2RML [6] and RML [7]. SHACL [43] is a W3C recommendation designed to validate the quality of RDF graphs against a set of conditions. However, the proposed approach was limited due to using a non-domain specific ontology, named the SHACL validation report vocabulary [27] which was designed to represent aspects related to the assessed data graphs, such as the subject and object of triples. However, the ontology does not contain suitable concepts and relationships to represent aspects of a mapping such as triple maps. Furthermore, blank nodes are reassigned a unique identifier in each validation report generated which reduced traceability due to the large number of blank nodes in mappings such as those represented in R2RML and RML. Moreover, SHACL does not support the update of graphs which is required for quality refinement of mappings. Thus, SPARQL [30] was required to change the mapping during the quality refinement process.

5.2. Lessons Learned from Evaluation Strategy

The following lessons were learned as a result of the research described in this article.

Mapping quality issues are common. Experiment 1 (See Section 4.2) discovered a mean of 8 quality issues per declarative uplift mapping tested. It is expected that these quality issues could have exponentially propagated into the resulting dataset after execution. In addition, similar work on mapping quality, such as [3] and [21] has identified a high number of mapping quality issues. These observations indicate that quality issues in mappings are common and could result in a decrease of quality in published linked data datasets. Thus, removing mapping quality issues will help to eradicate the root cause of many issues in datasets.

Diverse participant background is beneficial. Capturing feedback from non-experts (students) and domain experts (mapping and ontology design) provided diverse feedback which helped to identify improvements required for certain participants. In addition, it allowed the level of required background knowledge to be determine and hopefully reduced to improve uptake of the framework. The results of the evaluation strategy for the approach indicated that lower background knowledge influenced effectiveness.

Final Validation is important. OSCD and MQIO went through multiple iterations of refinement prior to the final validation. The iterations were guided by feedback from domain experts (mapping specialists) which helped to capture inconsistencies with the knowledge being represented. However, experts in

27 https://www.w3.org/TR/shacl/#validation-report
ontology design were required to ensure that inconsistencies were not introduced as a result of the changes in these iterations. Thus, validating of the resulting designs helped to determine the level of conformance with respective aspects, such as reuse of existing ontology design methodologies and application of relevant evaluation methods to the designs.

6. Conclusion

The evaluation strategy described in this article provides a method which collects feedback using various instruments, metrics and background knowledge. The experiments provided evidence to determine the perceived usability, effectiveness and understanding of the framework. In addition, the evidence allowed requirements to be validated through multiple sources of quantitative and qualitative data. It is hoped the methods could be reused by other researchers in order to validate similar linked data approaches. In addition, it is hoped the lessons learned as a result of applying the evaluation strategy could provide useful insights to guide researchers when completing user testing on their proposed approach.

Future work includes testing of the improvements outlined in this article. Thereafter, the improved framework will be published onto a public domain which will allow a larger and more diverse audience to use it. It is hoped increased uptake of the framework will result in useful feedback which can be used to continuously refine the design. In addition, the open-source implementation of the framework will promote contribution and collaboration from other researchers. Further relevant feedback will be taken into consideration and addressed when required.

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.

Special thanks to John Keeney & Liam Fallon at Ericsson Software Technology and Kevin Thompson & Ciara Beddy at Ordnance Survey Ireland for their expertise and support during the application case studies.

References


Appendix A. Snippet of RDF Turtle describing the quality assessment, refinement and validation of an uplift mapping in MQIO

```turtle
<osi-mapping.ttl> a mqio:MappingArtefact;
    mqio:wasCreatedBy ex:user-1;
    mqio:hasPurpose "Research Project";

ex:mappingQualityAssessment a mqio:MappingAssessment;
    mqio:assessedMapping <osi-mapping.ttl>;
    mqio:wasExecuted mqio:metric:D2;
    mqio:usedTool ex:mappingEditor;
    prov:wasAssociatedWith ex:user-1;
    prov:endedAtTime "2022-10-18T17:31:01.286820"^^xsd:dateTime;
    mqio:hasQualityRequirement ex:qualityRequirement;
    mqio:hasValidationReport ex:mappingValidationReport;
    mqio:hasViolation ex:violation-0.

ex:violation-0 a mqio:MappingViolation;
    mqio:hasLocation "predicateObjectMap-1";
    mqio:hasObjectValue geo:asWKT;
    mqio:inTripleMap <#TripleMap1>;
    mqio:isDescribedBy mqio:metric:D2;
    mqio:hasResultMessage "Usage of undefined property.".

ex:refinement-0 a mqio:MappingRefinement;
    mqio:usedQuery "";
    PREFIX rr: <http://www.w3.org/ns/r2rml#>
    DELETE { ?predicateObjectMap rr:predicate ?property }
    INSERT { ?predicateObjectMap rr:predicate
        <http://www.opengis.net/ont/geosparql#asWKT> } 
    WHERE { 
        SELECT ?predicateObjectMap ?property
            FILTER(str(?predicateObjectMap) = <BLANK_ID>).} 
    };
    mqio:hasConfidence "0.75"^^xsd:double;
    prov:endedAtTime "2022-10-18T17:31:01.286820"^^xsd:dateTime;
    prov:wasAssociatedWith ex:user-1;
    mqio:hasRefinementName "Find Similar Predicates";
    mqio:refinedViolation ex:violation-0.

ex:qualityRequirement a mqio:QualityRequirement;
    rdfs:comment "Quality requirement associated with the mapping.";
    mqio:isSatisfied "true"^^xsd:boolean;
```

Listing 1: Sample change report represented in MQIO

Appendix B. Snippet of RDF Turtle describing the changes in source of data of an uplift mapping in OSCD

```turtle
ex:changeLog-1
a oscd:ChangeLog ;
oscd:hasMaintainer ex:user-1 ;
oscd:hasDetectionStart "2022-11-00T00:00:00.0000000"^^xsd:dateTime ;
oscd:hasDetectionEnd "2022-12-31T00:00:00.0000000"^^xsd:dateTime ;
oscd:hasCurrentSource <https://raw.githubusercontent.com/kg-construct/rml-test-cases/master/test-cases/RMLTC0002a-JSON/student.json> ;
```
Listing 2: Sample change report represented in OSCD

Listing 3: SPARQL query to link source changes and quality issues with respective uplift mappings