CHAD ASK: Experimenting with a semi-automatic approach based on online surveys to formalise unstructured knowledge in Linked Data

Sebastian Barzaghi[1,*], Arianna Moretti[2,3], Ivan Heibi[2] and Silvio Peroni[2]

[1]University of Bologna, Department of Cultural Heritage, 40126 Bologna, Italy
[2]University of Bologna, Department of Classical Philology and Italian Studies (FICLIT), 40126 Bologna, Italy
[3]University of Helsinki, Computational History Group, 00014 Helsinki, Finland

Abstract

The growing adoption of computational methods in humanities and cultural heritage research necessitates aligning machine-readable representations with complex humanistic knowledge. Semantic Web technologies offer a robust framework for managing rich data relationships and enabling complex reasoning. However, the practical application of these technologies in the humanities remains limited due to the inherent technical specialisation required for modelling and mapping data, posing a significant barrier for non-technical domain experts. This paper addresses this gap by introducing CHAD-ASK, a novel plugin for Morph-KGC, a Python-based RML data conversion engine. CHAD-ASK translates the structured knowledge gathered from a LimeSurvey JSON response file into YARRRML mapping rules and Morph-KGC configuration files. Through the experience of a case study in converting data and metadata about a temporary exhibition into a Knowledge Graph, the tool successfully demonstrated its ability to generate the necessary artefacts to transform raw tabular survey data into fully compliant RDF triples, allowing researchers to perform complex metadata conversion without direct interaction with code or mapping languages. While CHAD-ASK significantly lowers the technical barrier to Semantic Web adoption, current limitations include the need for full assessment in standalone contexts, the integration of additional mapping possibilities to enhance semantic expressivity, and the future ability for users to extend the underlying data model via the tool itself. Future work will focus on systematic usability testing, enhancing quality control mechanisms, and integrating with additional tools for automated workflow execution.

Keywords

data mapping, data conversion, RML, Morph-KGC, RDF, YARRRML

1. Introduction

As computational methods become increasingly embedded in the practices of arts and humanities research, the challenge of aligning machine-readable representations with humanistic knowledge production becomes more pronounced [1]. Semantic Web (SW) technologies - like the Resource Description Framework (RDF) [2], ontologies [3], and Knowledge Graphs (KGs) [4] - offer powerful frameworks for addressing specific needs of the humanities, such as expressing rich data relationships, integrating heterogeneous sources, and supporting complex querying and reasoning [5].

However, the practical adoption of these technologies in humanities contexts remains limited, despite the growing interest in them. Several factors contribute to this lag in adoption [6]. SW technologies often require ad hoc data handling approaches that are typically accessible only to users with specialised technical expertise in areas like data modelling, programming, or knowledge engineering. However, as domain experts are increasingly expected to produce and publish their data in open, machine-actionable formats, there is a growing need to design KG construction tools that accommodate their often limited technical skills [7].

IRCDL 2026: 22nd Conference on Information and Research Science Connecting to Digital and Library Science, February 19-20, 2026, Modena, Italy
*Corresponding author.
sebastian.barzaghi@unibo.it (S. Barzaghi); arianna.moretti@unibo.it (A. Moretti); ivan.heibi@unibo.it (I. Heibi); silvio.peroni@unibo.it (S. Peroni)
0000-0002-0799-1527 (S. Barzaghi); 0000-0001-5486-7070 (A. Moretti); 0000-0001-5366-5194 (I. Heibi); 0000-0003-0530-4305 (S. Peroni)

In recent years, various approaches have been developed to simplify the complex process of KG construction, aiming to make it accessible to a broader range of users. These include solutions based on graphical user interfaces [8] as well as those leveraging mapping languages [9]. However, designing tools that are both comprehensive and adaptable to the wide variety of use cases remains a significant challenge. In particular, mapping languages - such as RDF Mapping Language (RML) [10] or its more user-friendly counterpart Yet Another RDF Mapping Language (YARRRML) [11] - facilitate the transformation of heterogeneous data formats (like CSV, JSON, XML, and relational databases) into RDF, aligning them with the semantics of a target ontology. They use formal translation rules, typically defined in a mapping document, to support data integration and transformation into a common semantic structure across diverse sources [12]. As data formats and use cases continue to diversify, a growing number of mapping languages have emerged, each offering unique features and syntactic variations. While this expands the range of possibilities, it also adds complexity to the selection process. Furthermore, many of these languages retain a level of technical complexity that more often than not poses a barrier for domain experts - particularly those from non-technical fields such as the humanities - making it unrealistic to expect them to create semantic mappings without additional support or tooling.

Survey-based data collection is a long-standing and widely accepted methodology in disciplines such as cultural heritage studies, sociology and linguistics [13]. Tools like Google Forms, Survey Monkey and LimeSurvey allow domain experts to design and administer complex questionnaires to capture rich and structured domain knowledge. Despite their prevalence, the outputs of such surveys (Excel, JSON, CSV, or XML formats) typically remain disconnected from broader semantic infrastructures. As it was already established, transforming this data into RDF generally requires proficiency in either (or even both) programming or declarative mapping languages. For many researchers in the humanities, this steep learning curve poses a significant obstacle, limiting the reuse, interoperability, and long-term preservation of valuable datasets.

In response to this gap, this paper introduces CHAD-ASK, a novel plugin for Morph-KGC, a Python-based RML data conversion engine that offers configuration capabilities in YARRRML and User-Defined Functions (UDFs) to further extend the code [14], designed to generate executable mapping and configuration documents from survey responses. CHAD-ASK takes as input a LimeSurvey response (exported as a file in JSON format) and produces two files: a YARRRML document containing mapping rules for the generation of the final KG, and an INI configuration file required by Morph-KGC. These artefacts can then be directly used to transform raw survey data into RDF triples, ready to be queried, linked, and integrated into broader KGs.

The rest of the paper is organised as follows. Section 2 illustrates the state of the art with respect to the existing mapping languages in the SW landscape (2.1), the user-friendly tools for enforcing semantic mapping (2.2) and the points of contact between questionnaires and SW (2.3). Section 3 delves into the proposed approach, focusing on its motivations (3.1), technical architecture (3.2), and workflow and components (3.3). Section 4 focuses on the case study in which we designed and developed the ideas at the basis of our approach: The Other Renaissance (4.1), a temporary exhibition of heterogeneous museum objects to be digitised into a FAIR Digital Twin that could be visualised and used in different ways on different devices and screens. Furthermore, we focus on three specific use cases that exemplify the use of CHAD-ASK for data mapping in the cultural heritage domain (4.2), and we discuss the lessons learnt from the case studies, providing an overview of CHAD-ASK's positioning with respect to the tools reviewed in Section 2.2 (4.3). Finally, Section 5 concludes the paper by addressing limitations and future developments of our approach.

2. Related work

In this section, we offer an overview on some of the most important works in the fields of semantic mapping languages (2.1), tools for RDF mapping (2.2) and transformation of survey data into RDF (2.3).

2.1. Semantic Mapping Languages

The conversion of RDF data from structured sources involves using different approaches that specify how source data should be converted into triples, ranging from developing and running scripts written in a programming language, up to using declarative mapping languages to drive the graph materialisation [15]. With respect to the latter, there is a wide and thriving ecosystem of semantic mapping languages that grew in recent years, as more data became accessible and new needs to manage and transform such data arose. Proposals such as R2O [16], an extensible and declarative language to describe mappings between relational database schemas and ontologies, paved the way towards the next mapping languages based on ontologies. RDB to RDF Mapping Language (R2RML) [17] is one of the foundational standards in this space. As a W3C recommendation, R2RML offers a powerful and expressive model that enables users to describe mappings from relational databases to RDF. However, it can be verbose and difficult for non-technical end-users to write by hand, and limiting for those needing to materialise RDF statements from other types of data sources. A generalisation of R2RML called RDF Mapping Language (RML) [10] was thus designed to accommodate other data formats such as CSV, JSON and XML. During the years, other extensions of R2RML and RML were proposed to tackle specific problems or gaps in the existing mapping landscape, such as incorporating mechanisms to define data transformation functions [18, 19, 20, 21] and extending the range of possible data sources, including NoSQL databases [15] and web services [22]. RML and its extensions are source-agnostic and extensible so that the same unified mappings can be defined over different input formats to achieve dataset consistency and richer interlinking of RDF resources. Despite that, RML syntax tends to be verbose and technical, making it difficult for non-programmers and domain experts to use. To address this problem, YARRRML [11] was created as a syntax based on YAML Ain't Markup Language (YAML) that compiles into RML to enable the writing of human-readable mapping rules and lower the entry barrier for semantic data modelling.

The SPARQL Protocol and RDF Query Language (SPARQL) has also been used as a basis to extend and develop mapping languages to generate RDF data. For example, XSPARQL [23] is a mapping language that integrates SPARQL with XQuery to allow querying XML and RDF data using the same framework and transforming data from one format into the other. SPARQL-Generate [24] is an extension of SPARQL that generates RDF from a set of documents in arbitrary formats. With a different approach, Façade-X [25] presents a generic meta-model and toolchain for RDF conversion of non-RDF data through a form-based declarative interface with abstractions over the complexity of RDF, mapping languages (like RML or SPARQL-Generate), allowing mappings to be defined through intuitive interfaces without changing the SPARQL syntax. Façade-X is implemented within the framework of SPARQL Anything and can access and transform heterogeneous data sources with efficiency and low implementation costs.

Other mapping languages not based on either RML or SPARQL have been designed and implemented as well. ShExML [26] is a mapping language based on Shape Expressions (ShEx) [27], which uses declarations about heterogeneous data sources combined with a set of shapes that specify how these should be mapped. D-REPR [28] is another language for mapping datasets to RDF, designed as a format-independent solution based on JSONPath expressions and YAML syntax. Another interesting solution based on YAML is YAMA Mapping Language (YAMAML) [29], a lightweight YAML-based language for generating RDF that adapts elements from Yet Another Metadata Application (YAMA) to map non-RDF data structures into RDF according to predefined application profiles.

2.2. User-Friendly Tools for RDF Mapping

As SW technologies develop, one of the main challenges is how to allow domain experts who may not have a deep technical background to create high-quality RDF data. Most traditional mapping languages, such as R2RML and RML, require knowledge of RDF concepts and syntactic precision; however, this presents a usability gap for many users in fields like the social sciences, cultural heritage, or citizen science, where domain experts are often not technically savvy enough to generate RDF data on their own. To address these usability gaps, various tools have been developed that offer graphical interfaces, mapping wizards, or form-based workflows to help guide users through generating RDF.

The majority of user-friendly RDF mapping environments have been designed around tabular data, addressing the practical need to convert spreadsheets into Linked Data. Early examples include RDF123 [30], which provides an application and a web service for writing mapping specifications and executing them against spreadsheets, supporting conditional and non-homogeneous row mappings. OpenRefine [31] is a well-known tool, particularly used in the Digital Humanities domain, that enables semi-automatic operations of data cleaning and reconciliation against external datasets. Odalic [32] interprets tabular data through column annotation, entity linking, and a user feedback loop for refining semantic mappings, exporting results as RDF datasets. Karma [33] allows users to import structured data and visually align them with ontologies to generate R2RML mappings that can be further refined through machine learning and ontology-based inference. Grafterizer [34] provides an environment for knowledge workers for data cleaning, transformation and semantic modelling via a graphical interface built on top of Grafter, a backend library and domain-specific language to specify transformations of data to RDF. Sheet2RDF [35] is a tool that enables users to convert annotated spreadsheets into RDF datasets, using templates and mappings between columns and ontology terms. Finally, an alternative approach for KG construction from spreadsheet data is proposed in [36], featuring a graphical user interface that enables bulk cell annotation and validation, multi-valued cells handling, style elements recognition, and traceability of the graph construction process.

Naturally, user-friendly tools for data mapping were developed also for other formats beyond tables. For example, the X3ML framework [37] is a suite of tools for defining mappings that regulate URI generation and data aggregation, transformation and integration between different XML data sources in a way that is accessible to domain experts. The framework leverages a series of software components, including a syntax normaliser, a mapping memory manager, schema validators and visualisers, an instance generator rule builder, a data transformation engine and X3ML, an XML-based mapping definition language that is used to establish matchings between a source and a target schema. All components have been designed for collaborative work, generalisability, process transparency and reuse of existing standards.

Moreover, there have been several proposals for tools and frameworks that support users in creating YARRRML mappings, including Matey [11], a browser-based graphical interface to help users create the rules in an interactive step-by-step manner by abstracting away the complexity of the underlying RML syntax and providing immediate validation against sample data, as well as export of the corresponding RML for inspection of the resulting RDF. Another tool worth mentioning is GRAPE [38], an open-source editor that allows RML mappings to be created with RDF schema extensions without losing flexibility through a language-oriented approach, where each module of RML is treated as a domain-specific language, and the text-like projection interface operates on the abstract syntax tree for syntactic correctness, auto-completion, and contextual guidance, thereby reducing the entry barrier for new users while still allowing for the complexity required by advanced semantic modelling. Finally, Yatter [39] is an open-source bi-directional translator that can transform from YARRRML to RML or R2RML, or vice versa, and thus helps interoperability between mapping ecosystems.

2.3. Bridging Questionnaire Data and the Semantic Web

Structured questionnaires and surveys are commonly used data collection tools across different domains - including social sciences, healthcare, citizen science, and public administration. The data they produce is often semantically rich but is typically stored in either proprietary formats or in a form that is not standardised in any meaningful way, which makes it less interoperable and reusable. To bridge this gap, the transformation of questionnaire responses into RDF has been explored as a topic, but it remains a somewhat underexplored area.

In the healthcare and epidemiology domains, mapping electronic forms or clinical questionnaires to ontologies has proven helpful in biomedical projects where form builders like REDCap [40] or OpenClinica [41] are used and data is exported for conversion into RDF using predefined mappings to clinical terminologies (*e.g.*, SNOMED CT [42] or HL7 FHIR [43]), emphasizing the necessity for semantic interoperability to aggregate and integrate data across studies. Similarly, citizen science platforms such as SPOTTERON [44] gather structured observational data using user-submitted forms.

Closer to the SW community, some tools aim to support the semantic modelling of survey instruments and their results. For instance, the Survey Ontology (SVO) [45] and the Data Documentation Initiative RDF (DDI-RDF) [46] provide vocabularies to describe survey structure, question design, and variable definitions in RDF. However, these models focus on the metadata of surveys rather than converting actual response data into RDF triples. An example of a use case of semantic lifting of questionnaire data in the social sciences domain based on DDI-RDF is illustrated in [47] using Python and a series of Jupyter Notebooks.

To date, few practical tools automate the generation of RDF mappings from survey exports. While platforms like LimeSurvey offer structured JSON output, there is a lack of user-friendly tools that can take such data and automatically generate mapping rules. The aforementioned projects hint at the potential of form-driven RDF generation, but they often focus on authoring rather than post-hoc mapping. In this context, CHAD-ASK represents a novel contribution: it fills a methodological and technical gap by automatically generating YARRRML mapping rules and configuration files from LimeSurvey questionnaire responses (exported as JSON), thereby enabling the seamless transformation of user-specified extensions of a data model into RDF within a data engineering pipeline based on Morph-KGC.

3. Introducing CHAD-ASK

CHAD-ASK is a software tool designed to facilitate the generation of semantic mappings for RDF data production. It translates structured input collected from domain experts through a form into formal artefacts that are compatible with the Morph-KGC data transformation engine. More specifically, its main aim is to semi-automatically produce YARRRML mapping rules and INI configuration files from JSON files containing the results of LimeSurvey questionnaires, thus increasing reproducibility and allowing technology access to non-expert users.

In this sense, CHAD-ASK acts as a bridge between human-centred data collection practices and machine-oriented data publishing workflows. Its design aims to lower the entry barrier for humanities scholars while translating their data into RDF triples expressed through one or more models. In this study, the model used was the Cultural Heritage Acquisition and Digitisation Application Profile (CHAD-AP) [48], a semantic data model developed with the Simplified Agile Methodology for Ontology Development (SAMOD) [49], expressed in OWL 2 DL and completely based on reuse of existing vocabularies, such as the CIDOC Conceptual Reference Model (CIDOC-CRM) [50], the Object-Oriented Library Reference Model (LRMoo) [51], the CRM Digital (CRMdig) [52] and the Getty's Art and Architecture Thesaurus (AAT) [53].

In the following sections, the conceptual ideation (3.1), the architecture (3.2) and the components (3.3) of CHAD-ASK are illustrated in more detail.

3.1. Theoretical Rationale

At the heart of CHAD-ASK lies a human-in-the-loop approach to KG construction. The tool is grounded in recognition of the fact that humanities data is often shaped by contextual and interpretive decisions not easily reducible to purely technical schemas. Forms and questionnaires, especially those administered through tools like LimeSurvey, are widely used across the humanities to structure field observations, metadata curation, and cataloguing processes. These instruments capture domain knowledge in a structured yet accessible form, making them ideal touchpoints for participatory modelling.

CHAD-ASK is primarily designed to serve two distinct target user profiles: the domain expert, who possesses deep knowledge of the metadata structure and content and aims to convert their dataset autonomously; and the digital humanist, who has expertise in data modelling and wrangling but lacks direct experience with RML and Morph-KGC as conversion technologies.

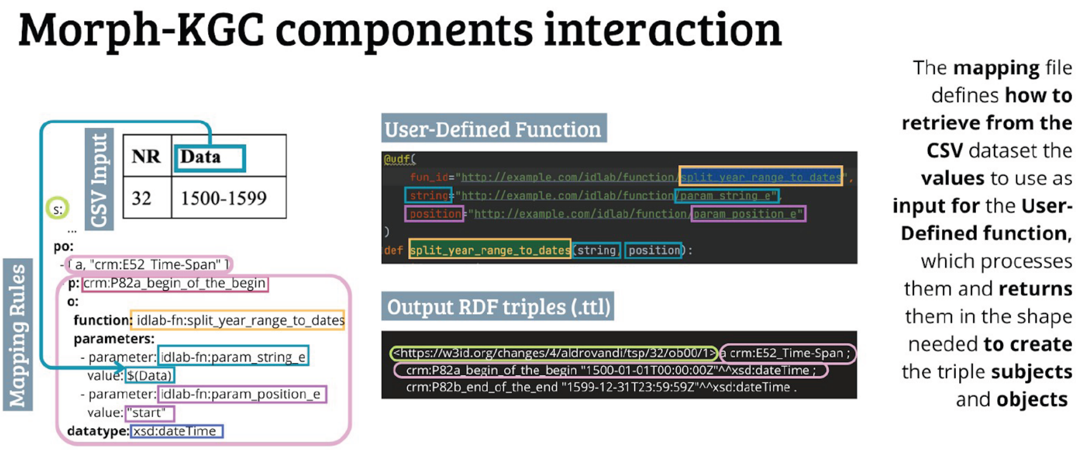

Figure 1: An image depicting the logic behind Morph-KGC components interaction, which will be then be automatically replicated thanks to the CHAD-ASK software extension. In the example, the mapping calls the function `split_year_range_to_date` using the value of the column “Data” (1500-1599) and the string “start” as input parameters, so as to assign the object `"1500-01-01T00:00:00Z"^^xsd:dateTime` to the predicate `P82a_begin_of_begin`.

Rather than requiring these users to master declarative mapping languages, CHAD-ASK reinterprets the mapping process as a series of conceptual questions that parallel the reasoning employed during data collection. This reframing aligns with epistemic values long emphasised in the humanities - transparency, contextualization, and interpretive openness. The mappings generated by CHAD-ASK are therefore not only technically functional but also semantically accountable. Each mapping decision can be traced back to the scholar’s explicit description of how their data should be understood and represented. The resulting mappings retain traceability to the scholars’ modelling intentions, making CHAD-ASK a conceptual scaffold for an open, reproducible semantic data pipeline.

3.2. Technical architecture

From a technical standpoint, CHAD-ASK is implemented as a plugin for the Morph-KGC framework [14], a Python-based system for RML-compliant data transformation (Figure 1). The integration with Morph-KGC offers several advantages: the use of a human-readable configuration format; extensibility through UDFs; comprehensive documentation; and an active developer community. These features collectively make it a suitable foundation for developing interfaces that mediate between expert knowledge and formal mapping expressions. While a basic familiarity with the underlying data model is still required to describe how implicit relations in the source data correspond to ontology classes and properties, CHAD-ASK significantly reduces the technical burden of producing executable mappings.

The tool was specifically designed to integrate with Morph-KGChad (Figure 2), an ad-hoc extension of the Morph-KGC engine developed to generate a KG structured according to CHAD-AP. Indeed, Morph-KGChad was initially designed to automatically convert tabular data strictly adhering to templates provided by the CHANGES Spoke 4 group for museum and digitisation process metadata (see Section 4.1). However, experience revealed limitations when dealing with input tables that are similar but not perfectly overlapping in format. Thus, CHAD-ASK initially stems from the need to address these weaknesses, aiming to semi-automatically manage the generation of new YARRRML mapping rules to interpret varied input data and reliably convert them into RDF. The prototype itself has been developed by reverse-engineering the logic of the YARRRML mapping used. This methodological choice primarily aims to ensure the management of future extensions of the model underneath, allowing the system to evolve in sync with changing data modelling needs. At the same time, CHAD-ASK aims to become a start pack builder for customised RDF conversion of datasets with different structures.

Input data (the survey’s results) are received in the form of JSON-encoded responses exported from LimeSurvey. Each questionnaire is organised in four main sections with the following purposes: (1) to

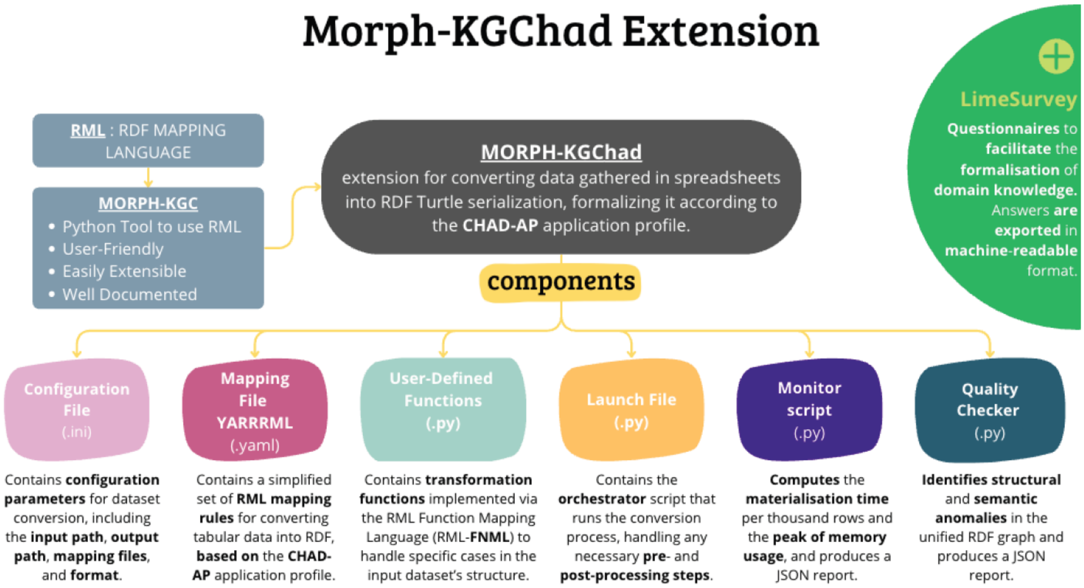

Figure 2: A diagram representing the Morph-KGChad extension and its components.

set up a configuration file; (2) to make the internal filling conventions of a data structure into mapping rules, by linking the mapping necessities to existing conversion functions; (3) to help formalise the data modelling necessities; and (4) to translate them into YARRRML rules. Although the survey is structurally defined, it can be generally adapted to different datasets through a set of questions that guide the user in specifying which are the fields to be considered in the data transformation process and, for each field, which is the property or semantic pattern to be mapped to it. Once completed and generated as JSON data, the questionnaire is processed by CHAD-ASK to produce two primary outputs: a .yaml file containing YARRRML mapping rules for one or more data sources, and a .ini configuration file specifying the Morph-KGC configuration necessary for RDF generation within the Morph-KGC environment. The process has been designed considering that each case study comes with specific necessities, thus including the possibility for the users to intervene on the template if necessary, and to allow a domain expert without a SW background to formalise domain expertise.

3.3. Workflow and Components

The generation process is structured into four main phases, each corresponding to a section of the LimeSurvey questionnaire.

Configuration Setup. In the first phase, global parameters required for RDF generation are gathered. Users specify settings such as output directories, serialisation formats (*e.g.*, Turtle or JSON-LD), handling of null values, logging preferences, and base IRI definitions. Additional options cover project versioning, character encoding, and datatype inference. These parameters are automatically compiled into the [CONFIGURATION] section of the Morph-KGC .ini file. The questionnaire provides explanatory guidance and default suggestions, ensuring that users can make informed decisions without needing to consult technical documentation.

Internal Data Conventions. The second phase concerns assumptions about the structure and format of the input data. Here, users define one or more data source blocks - corresponding to Morph-KGC's [DataSourceX] sections - where they can specify file paths, delimiters, encodings, and partitioning strategies. By formalising such conventions explicitly, CHAD-ASK ensures that Morph-KGC can interpret heterogeneous source data correctly, thus minimising potential errors in parsing and transformation.

Model Extension and Mapping Formalisation. The questionnaire is structured, but it can be adapted to different datasets through a set of questions guiding the user in identifying the relevant fields for the conversion and, for each field, the property or semantic pattern to be applied. We distinguish two recurring demands that impact both the generalisability and the level of semantic expressiveness supported by the questionnaire:

A) **RDF materialisation bound to Chad-AP.** This basic-use modality is adequate when the user wants to convert a tabular dataset into RDF while remaining within a reference profile (in our case, CHAD-AP or a CIDOC-CRM-compliant model). In this mode, the questionnaire is mainly focused on property-level specification: redirecting the input dataset structure into the CHAD-AP logic, suitable for most GLAM datasets, so that it can be converted by exploiting Morph-KGChad functionalities. The user is required to provide basic answers meant to interpret the structure of their input data, to associate columns with existing predicates, thus choosing from a controlled set of modelling choices.

B) **Specification at the semantic pattern level.** The advanced mode, intended to be exploited by users who are experts in semantic modelling but not in RML, allows for richer modelling choices by specifying a semantic pattern, understood as a structured combination of classes and properties, as well as indications about URI structuring. This mode increases generalisability to different domains and the adoption of data models other than CHAD-AP, but necessarily requires greater modelling awareness.

Even in basic use, it is possible to introduce extensions to the logic of the profile, such as adding a pattern to extend a representation compliant with CHAD-AP (or CIDOC-CRM in general), which is formalised by CHAD-ASK following the operational logic managed by Morph-KGChad. If, on the other hand, the extension requirement is complex enough to go beyond the logic implemented by CHAD-AP, it is possible to integrate the logic of the advanced mode (B) to manage the specific case, while still using Morph-KGChad's RDF materialisation methods for the rest of the dataset. When the user declares from the beginning the intent to operate as an expert user and adopt their own data model for the materialisation, the questionnaire branches directly to mode (B), with mode (A) being optional.

In both cases, CHAD-ASK produces the necessary documents to run in Morph-KGC (configuration files and mapping rules). If the input dataset can be fully represented using CHAD-AP, the files generated by the questionnaire answers will just integrate and/or overwrite the existing ones already included in the Morph-KGChad software extension.

YARRRML Translation. Finally, the semantic mapping information collected in the previous phases is compiled into a valid YARRRML .yaml file. This file defines subject templates, predicate-object pairs, and any conditional or transformation logic necessary to produce meaningful RDF triples. The resulting mappings can be executed directly within the Morph-KGC pipeline, reused across different datasets, or extended as new data collection campaigns evolve.

4. Case study

In this section, we present a case study serving as the context in which we designed, developed and tested the tools and processes illustrated so far (4.1). Then, we illustrate three simple use cases that can be addressed through our approach (4.2). Finally, we make a comparative discussion between CHAD-ASK and some of the tools illustrated in Section 2.2 (4.3).

4.1. The Other Renaissance

Our work is set within the context of the Cultural Heritage Active Innovation For Next-Gen Sustainable Society (CHANGES) initiative, funded by the European Union and articulated across nine Spokes, aimed at promoting and preserving Italian cultural heritage. Specifically, this use case was developed within Spoke 4, which focused on the use of virtual technologies to promote, preserve and enhance cultural heritage objects housed in museums and art collections. The case study in question involved the complete digitisation of the temporary exhibition "The Other Renaissance: Ulisse Aldrovandi and the Wonders of the World" (held in Bologna during 2022-2023), with the goal of creating an interactive and scientifically reproducible Digital Twin of it. The process of digitising a whole heterogeneous collection of different objects (including written materials, scientific tools, specimens, and archaeological finds) was laden with challenges, ranging from technical complexities related to digitally acquiring objects stored within reflective glass cases, to logistics and legal requirements with managing certain objects' data [54]. From this perspective, one of the main methodological challenges was the conversion of heterogeneous data - consisting of both descriptive bibliographic metadata of the cultural heritage objects and paradata related to their digitisation - from source data, organised in a tabular format, into Linked Open Data.

The key outcome of this process was the development of the Cultural Heritage Acquisition and Digitisation Knowledge Graph (CHAD-KG) [55], an RDF dataset that serves as the primary source of descriptive and procedural data for the objects within the Digital Twin. To ensure data accessibility and reusability, a complete dump of CHAD-KG was published on Zenodo [56]. Furthermore, to facilitate exploration by non-specialist users, a navigable version was made available via a lightweight static website, generated automatically from the KG itself via the Static Publisher of Knowledge software (SPUK) [57].

4.2. CHAD-ASK use cases

The current demo version is meant to represent, in a nutshell, all the working logic behind CHAD-ASK and its integration within Morph-KGChad. The current implementation is a prototype meant to put into practice a representative subset of the questionnaire's logic, focusing on the interaction between (1) questionnaire responses (exported in JSON format), (2) generated files (INI configurations and YARRRML rules) and (3) the CHAD-ASK software component, to represent their execution in the Morph-KGChad pipeline. Although the questionnaire design supports both property selection and pattern specification, in line with its generalisable purpose, the demo currently implements a limited number of reusable patterns selected based on the requirements that emerged during the RDF materialisation of the case study. For this purpose, three use-cases were selected to provide an overview of the survey's structure and its output in case one or more functions are called individually, sequentially, or in a nested way. In addition to that, the demo also includes the software components meant to (1) convert the JSON output of the questionnaire into mapping rules (.yaml) and configuration files (.ini), and (2) to combine the output with the existing codebase, either by expanding the information or overwriting it.

The code of the demo is available on GitHub (https://github.com/dharc-org/morph-kgc-changes-metadata/tree/main/src/ask-kg), with input and output materials used in the demo development and testing. The LimeSurvey questionnaire has been published on Zenodo [58]. Below, the three case studies are documented.

Case 1: Guided overwriting of a minimal subset of configuration parameters. The first case study focuses on the generation of the configuration.ini file. In the questionnaire, the user is asked some basic questions regarding (1) the paths of the input dataset CSV files and mapping files, (2) the path where the output file will be generated, (3) the RDF serialisation to be chosen among a series of supported formats, (4) the version number of the output dataset, and (5) the base path of the IRIs to be generated. The information extraction process is very simple, and the code for managing this step (`src/ask-kg/demo/ini_config_demo1.py`) is mainly designed to automate the updating of the configuration file, overwriting only a few specific fields of the precompiled configuration file provided in Morph-KGChad.

Case 2: Developing mapping rules for time-span-related triples generation. On a more complex level, the second case concerns the generation of a mapping rule involving the use of non-nested user-defined functions. The purpose is to produce the triples concerning the start and end dates of the artefact's timespan. The degree of complexity is limited because, as the type of data is pre-defined (dates), some of the information can be automatically asserted by the system. The only information that the system needs to take as input from the user is as follows: (1) the title of the column containing the information relating to the date of creation or origin of the object, (2) whether the value can be a single date only or also a range, (2.1) in the case of time ranges, whether the two dates can be separated by characters other than "-". If so, the user is asked to specify other separators used. The answers to these three questions is then exported in JSON and interpreted by the CHAD-ASK software component (`src/ask-kg/yarrrml_object_demo2_dates.py`), aimed at extracting the three values and using them as input of the mapping. In this case, there is no need to specify the nature of the field value, because it is implicit that it is a date. The functions involved in the process, `normalise_and_convert_to_iri` and `split_year_range_to_date`, operate independently, which means that they are called directly on the input data extracted from the table and not on the output of another function.

Case 3: Developing mapping rules for Acquisition Tools IRI generation. The last case study (described in Figure 3) involves the use of nested user-defined functions, with the aim of assigning an IRI to the tools used for acquisitions in the digitisation process. In this case, there are more variables, and the user must specify (1) the title of the column containing the information relating to the acquisition equipment, (2) whether multiple values are allowed in the CSV cell, (2.1) if so, which separators are used, and - given that IRI generation is a cross-cutting task in the dataset - (3) the nature of the content of the column. In this case, as technical equipment is concerned, the answer to be selected from the drop-down menu will be "electronic device". The information will be converted into an identifying label in the generation of the IRI itself. The peculiarity of this case study managed by CHAD-ASK (`src/ask-kg/yarrrml_process_demo3_iris.py`) lies in the use of nested functions: the input value of the `normalise_and_convert_to_iri` function is the output of the `multiple_separator_split_explode` function, which is used to separate multiple values in the same cell.

4.3. Comparative Discussion

Experimenting with the CHAD-ASK approach in the case studies provided the opportunity to highlight some limitations and strengths of the resource, in comparison with some of the user-friendly mapping tools introduced among the related works, whose aims are only partially overlapping.

From an initial comparison, it appears that, unlike many of the tools discussed in Section 2.2, CHAD-ASK is not intended as an interactive environment for live mapping authoring. Indeed, in most cases, these tools offer guided interfaces to familiarise users with schemas by annotating columns and values. Among the tools oriented to the generation of mappings, Karma produces R2RML rules, visually supporting alignment with existing ontologies, while Matey guides the user in the design of YARRRML rules, offering the option to export the resulting file. Among the tools designed primarily for generating RDF data (rather than for producing a reusable mapping document), Odalic and Sheet2RDF adopt a table-driven approach, in which the annotation of the data structure guides the transformation and final export to RDF serialisations. In contrast, OpenRefine mainly offers upstream data cleaning and harmonisation tasks.

Unlike the examples listed above, CHAD-ASK is located in a preliminary step of RDF materialisation, in which domain knowledge is formalised through a structured collection of modelling decisions conveyed by a questionnaire in natural language, common in humanities workflows, hiding the code and mapping languages from users who may not have technical skills. In practice, the Morph-KGC CHAD-ASK plugin converts the responses exported in JSON from LimeSurvey into INI configuration files and YARRRML mapping rules, maintaining explicit traceability between user choices and generated

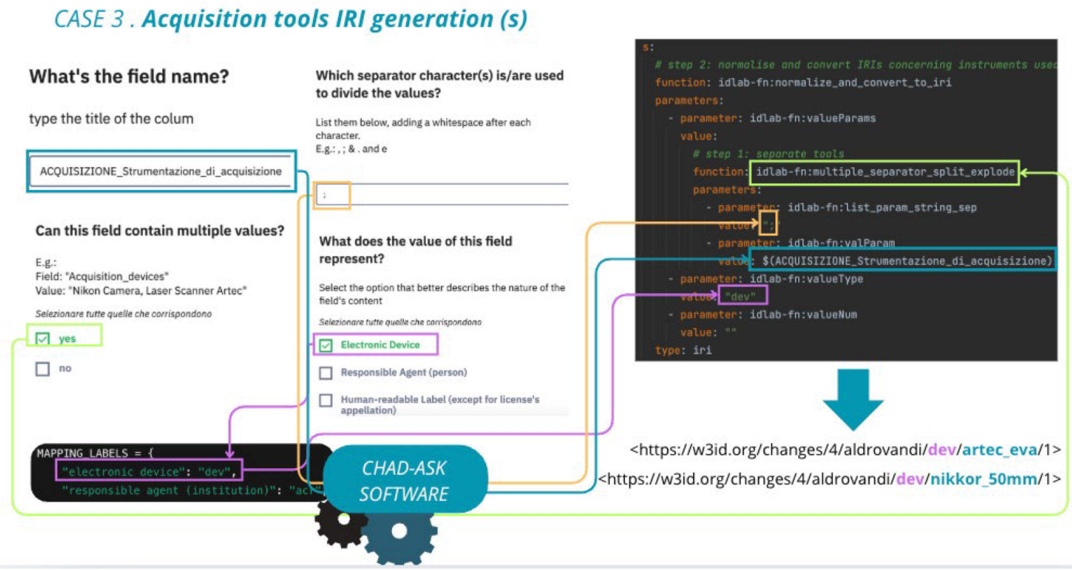

Figure 3: Representation of the mechanism behind the management of the third case study, assumed at the acquisition tools IRI generation. The image depicts how the LimeSurvey answers are converted into mapping parameters to generate IRIs.

values. This makes it a particularly suitable tool when the primary objective is to obtain RDF data from tables with guided control, significantly reducing the need for technical assistance.

The case studies presented show that, when using the resource in its basic application, the adoption of CHAD-ASK reduces the amount of authoring required for mapping by limiting the decision-making space to a controlled set of properties and patterns. In this case, the options are already modelled in CHAD-AP and managed by the Morph-KGChad extension: this comes with the advantage of ensuring the materialisation of semantic data according to a standardised pipeline starting from GLAM tabular datasets, even if gathered following different conventions in terms of delimiters, separators, persistent identifiers and value normalisation. In advanced applications, CHAD-ASK also serves as a tool for generating reusable mapping and configuration files ready for Morph-KGC, not modelled on CHAD-AP, and thus supporting more expert use by data modelling professionals, who can specify new patterns for greater control over semantic expressiveness.

It's worth noting that CHAD-ASK does not aim to replace a comprehensive visual environment for ontology alignment, but is rather intended to complement such approaches by leveraging the contributions of domain experts without technical skills and modelling experts without prior knowledge of RML logic.

5. Conclusion

Within the larger discussion about the technical complexity involved in KG construction, this paper addressed the gap between the growing necessity for domain experts to publish machine-actionable data and their often limited technical expertise in data modelling and mapping languages. We introduced CHAD-ASK, a novel tool designed to generate executable mapping and configuration documents from survey responses. The core idea behind CHAD-ASK is to leverage the familiar paradigm of questionnaires to formalise the domain knowledge of a user without requiring any technical background. By establishing the degree of freedom in the type of answer that can be given, CHAD-ASK effectively guides non-technical users through the complex process of semantic alignment between an existing data model and the data to be materialised into a KG. Thus, CHAD-ASK acts as an intermediary for Morph-KGC, generating the necessary YARRRML mapping rules and configuration files directly from a LimeSurvey JSON response file. The generated artefacts can then be used to transform raw survey data (typically CSV or similar flat formats) into fully compliant RDF triples. The demonstrated use cases successfully illustrated how researchers can perform complex metadata conversion from CSV to RDF without ever having to directly interact with code or declarative mapping rules.

While effective, the current version of this approach still shows a number of limitations. First, the current reusability of the framework has not been fully evaluated in standalone contexts. Assessment is required to determine the ease of adopting the use of CHAD-ASK in external projects relying only on its built-in functions. Second, the full exploitation of all CHAD-AP modelling scenarios and Morph-KGChad mapping possibilities is not yet completed, thus restricting the maximal expressivity of the resulting knowledge graph. A thorough testing of the model and the extension is needed to assess their adaptability to broader or different domains beyond those considered in the pilot study. Third, in order to adapt the questionnaire and the data mapping to the specific needs that can vary from project to project, it would be ideal to give the user the possibility to extend the data model underneath via CHAD-ASK itself.

Future work is planned to address these limitations and, in addition, to enhance the automation and quality control mechanisms of the data pipeline. In particular, we plan to systematically test CHAD-ASK and the other tools involved in our workflow (e.g., CHAD-AP) for domain extensions, and to explore whether the proposed approach could be applied in other disciplines, such as lexicography. This includes the development of additional case studies designed exclusively to evaluate the reusability of CHAD-ASK and CHAD-AP in external, standalone projects. In addition to that, to demonstrate the robustness of the presented resource, a user-centred evaluation will be integrated through user studies (including questionnaires and task-based testing), combining objective metrics (e.g., completion time and number of clicks, possibly compared with alternative tools) and subjective metrics (based on qualitative questions about perceived ease of use), to empirically verify that users are successfully able to generate semantic mappings with CHAD-ASK. Further, work will be finalised to complete the integration of all Morph-KGChad mapping possibilities, ensuring the resulting knowledge graph achieves more semantic depth. To enhance data quality assurance, we will prioritise the addition of automated error-handling reporting for data inconsistencies. This feature will provide immediate feedback on data quality issues arising during the transformation process. Finally, one interesting possible development towards automatic code component execution revolves around the integration between CHAD-ASK and MITAO [59], a tool for automatically executing and reproducing text analysis workflows.

Acknowledgments

This work was partially funded by Project PE 0000020 CHANGES - CUP B53C22003780006, NRP Mission 4 Component 2 Investment 1.3, funded by the European Union - NextGenerationEU.

Declaration on Generative AI

During the preparation of this work, the authors used Grammarly and DeepL in order to: Text translation and Grammar and spelling check. Furthermore, ChatGPT (GPT-5) was used in a limited way in order to: Paraphrase and reword certain sentences for improved clarity.

References

[1] B. Joyeux-Prunel, Digital humanities in the era of digital reproducibility: towards a fairest and post-computational framework, International Journal of Digital Humanities 6 (2024) 23–43. doi:10.1007/s42803-023-00079-6.

[2] S. Decker, S. Melnik, F. van Harmelen, D. Fensel, M. Klein, J. Broekstra, M. Erdmann, I. Horrocks, The Semantic Web: the roles of XML and RDF, IEEE Internet Computing 4 (2000) 63–73. doi:10.1109/4236.877487.

[3] M. Uschold, M. Gruninger, Ontologies: principles, methods and applications, The Knowledge Engineering Review 11 (1996) 93–136. doi:10.1017/S0269888900007797.
[4] A. Hogan, E. Blomqvist, M. Cochez, C. D'Amato, G. De Melo, C. Gutierrez, S. Kirrane, J. E. Labra Gayo, R. Navigli, S. Neumaier, A.-C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, A. Zimmermann, Knowledge Graphs, ACM Computing Surveys 54 (2021) 71:1–71:37. doi:10.1145/3447772.
[5] N. Shadbolt, T. Berners-Lee, W. Hall, The Semantic Web Revisited, IEEE Intelligent Systems 21 (2006) 96–101. doi:10.1109/MIS.2006.62.
[6] A. Hogan, The Semantic Web: Two decades on, Semantic Web 11 (2020) 169–185. doi:10.3233/SW-190387.
[7] L. Vogt, K. E. Farfar, P. Karanth, M. Konrad, A. Oelen, M. Prinz, P. Stroemert, Rosetta Statements: Simplifying FAIR Knowledge Graph Construction with a User-Centered Approach, 2025. doi:10.48550/arXiv.2407.20007.
[8] X. He, Y. Liu, Research on the Design and Implementation of a Knowledge Graph Construction Tool, in: Proceedings published by Atlantis Press, 2023, pp. 605–615. doi:10.2991/978-94-6463-238-5_82.
[9] J. Arenas-Guerrero, M. Scrocca, A. Iglesias-Molina, J. Toledo, L. P. Gilo, D. Doña, O. Corcho, D. Chaves-Fraga, Knowledge Graph Construction with R2RML and RML: An ETL System-based Overview, 2021. URL: https://openreview.net/forum?id=vz5u6Ji1Gkd, accessed December 11, 2025.
[10] A. Dimou, M. Vanden Sande, P. Colpaert, Rml: A generic language for integrated rdf mappings of heterogeneous data, 2014.
[11] P. Heyvaert, B. De Meester, A. Dimou, R. Verborgh, Declarative rules for linked data generation at your fingertips!, in: A. Gangemi, A. L. Gentile, A. G. Nuzzolese, S. Rudolph, M. Maleshkova, H. Paulheim, J. Z. Pan, M. Alam (Eds.), The Semantic Web: ESWC 2018 Satellite Events, Springer International Publishing, Cham, 2018, pp. 213–217. doi:10.1007/978-3-319-98192-5_40.
[12] A. Iglesias Molina, Knowledge Graph Construction and Evolution using Declarative Mapping Languages, Ph.D. thesis, Universidad Politécnica de Madrid, 2024. doi:10.20868/UPM.thesis.83228.
[13] R. K. Kelsall, A. Poole, A. Kuhn, The Questionnaire in a Sociological Research Project, The British Journal of Sociology 23 (1972) 344–357. doi:10.2307/589082.
[14] J. Arenas-Guerrero, D. Chaves-Fraga, J. Toledo, M. S. Pérez, O. Corcho, Morph-KGC: Scalable knowledge graph materialization with mapping partitions, Semantic Web 15 (2024) 1–20. doi:10.3233/SW-223135.
[15] A. Iglesias-Molina, A. Cimmino, E. Ruckhaus, D. Chaves-Fraga, R. García-Castro, O. Corcho, An ontological approach for representing declarative mapping languages, Semantic Web 15 (2024) 191–221. doi:10.3233/SW-223224.
[16] J. Barrasa Rodríguez, O. Corcho, A. Gómez-Pérez, R2O, an extensible and semantically based database-to-ontology mapping language, in: Proceedings of the Second Workshop on Semantic Web and Databases, SWDB 2004, Springer-Verlag, 2004, pp. 1069–1070. URL: http://www.springer.com/computer/database+management+%26+information+retrieval/book/978-3-540-24576-6, accessed December 11, 2025.
[17] M. Rodríguez-Muro, M. Rezk, Efficient SPARQL-to-SQL with R2RML mappings, Journal of Web Semantics 33 (2015) 141–169. doi:10.1016/j.websem.2015.03.001.
[18] C. Debruyne, D. O'Sullivan, R2RML-F: Towards Sharing and Executing Domain Logic in R2RML Mappings, 2016.
[19] B. De Meester, W. Maroy, A. Dimou, R. Verborgh, E. Mannens, Declarative Data Transformations for Linked Data Generation: The Case of DBpedia, in: E. Blomqvist, D. Maynard, A. Gangemi, R. Hoekstra, P. Hitzler, O. Hartig (Eds.), The Semantic Web, Springer International Publishing, Cham, 2017, pp. 33–48. doi:10.1007/978-3-319-58451-5_3.
[20] A. C. Junior, C. Debruyne, R. Brennan, D. O'Sullivan, FunUL: a method to incorporate functions into uplift mapping languages, in: Proceedings of the 18th International Conference on Information Integration and Web-Based Applications and Services, Association for Computing Machinery,

New York, NY, USA, 2016, pp. 267–275. doi:10.1145/3011141.3011152.
[21] J. Slepicka, C. Yin, P. Szekely, C. A. Knoblock, KR2RML: An Alternative Interpretation of R2RML for Heterogeneous Sources?, 2015.
[22] A. Chortaras, G. Stamou, D2RML: Integrating Heterogeneous Data and Web Services into Custom RDF Graphs, 2018.
[23] S. Bischof, S. Decker, T. Krennwallner, N. Lopes, A. Polleres, Mapping between RDF and XML with XSPARQL, Journal on Data Semantics 1 (2012) 147–185. doi:10.1007/s13740-012-0008-7.
[24] M. Lefrançois, A. Zimmermann, N. Bakerally, A SPARQL Extension for Generating RDF from Heterogeneous Formats, in: E. Blomqvist, D. Maynard, A. Gangemi, R. Hoekstra, P. Hitzler, O. Hartig (Eds.), The Semantic Web, Springer International Publishing, Cham, 2017, pp. 35–50. doi:10.1007/978-3-319-58068-5_3.
[25] E. Daga, L. Asprino, P. Mulholland, A. Gangemi, Facade-X: An Opinionated Approach to SPARQL Anything, in: Further with Knowledge Graphs, IOS Press, 2021, pp. 58–73. doi:10.3233/SSW210035.
[26] H. García-González, I. Boneva, S. Staworko, J. E. Labra-Gayo, J. M. Cueva Lovelle, ShExML: improving the usability of heterogeneous data mapping languages for first-time users, PeerJ Computer Science 6 (2020) e318. doi:10.7717/peerj-cs.318.
[27] E. Prud'hommeaux, J. E. Labra Gayo, H. Solbrig, Shape expressions: an RDF validation and transformation language, in: Proceedings of the 10th International Conference on Semantic Systems, Association for Computing Machinery, New York, NY, USA, 2014, pp. 32–40. doi:10.1145/2660517.2660523.
[28] B. Vu, J. Pujara, C. A. Knoblock, D-REPR: A Language for Describing and Mapping Diversely-Structured Data Sources to RDF, in: Proceedings of the 10th International Conference on Knowledge Capture, Association for Computing Machinery, New York, NY, USA, 2019, pp. 189–196. doi:10.1145/3360901.3364449.
[29] N. Thalhath, M. Nagamori, T. Sakaguchi, YAMAML: An Application Profile Based Lightweight RDF Mapping Language, in: Y.-H. Tseng, M. Katsurai, H. N. Nguyen (Eds.), From Born-Physical to Born-Virtual: Augmenting Intelligence in Digital Libraries, Springer International Publishing, Cham, 2022, pp. 412–420. doi:10.1007/978-3-031-21756-2_32.
[30] L. Han, T. Finin, C. Parr, J. Sachs, A. Joshi, RDF123: From Spreadsheets to RDF, in: A. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. Finin, K. Thirunarayan (Eds.), The Semantic Web – ISWC 2008, Springer, Berlin, Heidelberg, 2008, pp. 451–466. doi:10.1007/978-3-540-88564-1_29.
[31] K. Ham, OpenRefine (version 2.5). http://openrefine.org. Free, open-source tool for cleaning and transforming data, Journal of the Medical Library Association 101 (2013) 233–234. doi:10.3163/1536-5050.101.3.020.
[32] T. Knap, Towards Odalic, a Semantic Table Interpretation Tool in the ADEQUATe Project?, 2017.
[33] C. A. Knoblock, P. Szekely, J. L. Ambite, A. Goel, S. Gupta, K. Lerman, M. Muslea, M. Taheriyan, P. Mallick, Semi-automatically Mapping Structured Sources into the Semantic Web, in: E. Simperl, P. Cimiano, A. Polleres, O. Corcho, V. Presutti (Eds.), The Semantic Web: Research and Applications, Springer, Berlin, Heidelberg, 2012, pp. 375–390. doi:10.1007/978-3-642-30284-8_32.
[34] D. Sukhobok, N. Nikolov, A. Pultier, X. Ye, A. Berre, R. Moynihan, B. Roberts, B. Elvesæter, N. Mahasivam, D. Roman, Tabular Data Cleaning and Linked Data Generation with Grafterizer, in: H. Sack, G. Rizzo, N. Steinmetz, D. Mladenić, S. Auer, C. Lange (Eds.), The Semantic Web, Springer International Publishing, Cham, 2016, pp. 134–139. doi:10.1007/978-3-319-47602-5_27.
[35] M. Fiorelli, T. Lorenzetti, M. T. Pazienza, A. Stellato, A. Turbati, Sheet2RDF: a Flexible and Dynamic Spreadsheet Import&Lifting Framework for RDF, in: M. Ali, Y. S. Kwon, C.-H. Lee, J. Kim, Y. Kim (Eds.), Current Approaches in Applied Artificial Intelligence, Springer International Publishing, Cham, 2015, pp. 131–140. doi:10.1007/978-3-319-19066-2_13.
[36] M. Schröder, C. Jilek, M. Schulze, A. Dengel, Interactively Constructing Knowledge Graphs from Messy User-Generated Spreadsheets, 2021. doi:10.48550/arXiv.2103.03537.
[37] Y. Marketakis, N. Minadakis, H. Kondylakis, K. Konsolaki, G. Samaritakis, M. Theodoridou, G. Flouris, M. Doerr, X3ML mapping framework for information integration in cultural her-

itage and beyond, International Journal on Digital Libraries 18 (2017) 301–319. doi:10.1007/s00799-016-0179-1.
[38] J. Duchateau, C. Debruyne, GRAPE: Guiding RML Authoring with a Projectional Editor, in: Proceedings published by CEUR-WS.org, 2025. URL: https://orbi.uliege.be/handle/2268/332525, accessed December 11, 2025.
[39] A. Iglesias-Molina, D. Chaves-Fraga, I. Dasoulas, A. Dimou, Human-Friendly RDF Graph Construction: Which One Do You Chose?, in: I. Garrigós, J. M. Murillo Rodríguez, M. Wimmer (Eds.), Web Engineering, Springer Nature Switzerland, Cham, 2023, pp. 262–277. doi:10.1007/978-3-031-34444-2_19.
[40] E. F. Patridge, T. P. Bardyn, Research Electronic Data Capture (REDCap), Journal of the Medical Library Association 106 (2018) 142–144. doi:10.5195/jmla.2018.319.
[41] M. Cavelaars, J. Rousseau, C. Parlayan, S. de Ridder, A. Verburg, R. Ross, G. R. Visser, A. Rotte, R. Azevedo, J.-W. Boiten, G. A. Meijer, J. A. M. Belien, H. Verheul, OpenClinica, Journal of Clinical Bioinformatics 5 (2015) S2. doi:10.1186/2043-9113-5-S1-S2.
[42] L. Bos, Medical and Care Compunetics 3, IOS Press, 2006.
[43] D. Bender, K. Sartipi, HL7 FHIR: An Agile and RESTful approach to healthcare information exchange, in: Proceedings of the 26th IEEE International Symposium on Computer-Based Medical Systems, 2013, pp. 326–331. doi:10.1109/CBMS.2013.6627810.
[44] H.-Y. Liu, D. Dörler, F. Heigl, S. Grossberndt, Citizen Science Platforms, in: K. Vohland, A. Land-Zandstra, L. Ceccaroni, R. Lemmens, J. Perelló, M. Ponti, R. Samson, K. Wagenknecht (Eds.), The Science of Citizen Science, Springer International Publishing, Cham, 2021, pp. 439–459. doi:10.1007/978-3-030-58278-4_22.
[45] I. Baroni, I. Celino, D. Scandolari, G. Re Calegari, M. Scrocca, The Survey Ontology: Packaging Survey Research as Research Objects, 2021. doi:10.4126/FRL01-006429412.
[46] T. Bosch, R. Cyganiak, DDI-RDF Discovery Vocabulary: A Metadata Vocabulary for Documenting Research and Survey Data, 2013.
[47] L. Heling, F. Bensmann, B. Zapilko, M. Acosta, Y. Sure-Vetter, Building Knowledge Graphs from Survey Data: A Use Case in the Social Sciences (Extended Version), in: P. Hitzler, S. Kirrane, O. Hartig, V. de Boer, M.-E. Vidal, M. Maleshkova, S. Schlobach, K. Hammar, N. Lasierra, S. Stadtmüller, K. Hose, R. Verborgh (Eds.), The Semantic Web: ESWC 2019 Satellite Events, Springer International Publishing, Cham, 2019, pp. 285–299. doi:10.1007/978-3-030-32327-1_48.
[48] S. Barzaghi, I. Heibi, A. Moretti, S. Peroni, Developing Application Profiles for Enhancing Data and Workflows in Cultural Heritage Digitisation Processes, in: G. Demartini, K. Hose, M. Acosta, M. Palmonari, G. Cheng, H. Skaf-Molli, N. Ferranti, D. Hernández, A. Hogan (Eds.), The Semantic Web – ISWC 2024, Springer Nature Switzerland, Cham, 2025, pp. 197–217. doi:10.1007/978-3-031-77847-6_11.
[49] S. Peroni, A Simplified Agile Methodology for Ontology Development, in: M. Dragoni, M. Poveda-Villalón, E. Jimenez-Ruiz (Eds.), OWL: Experiences and Directions – Reasoner Evaluation, Springer International Publishing, Cham, 2017, pp. 55–69. doi:10.1007/978-3-319-54627-8_5.
[50] M. Doerr, The CIDOC CRM, an Ontological Approach to Schema Heterogeneity, in: Y. Kalfoglou, M. Schorlemmer, A. Sheth, S. Staab, M. Uschold (Eds.), Semantic Interoperability and Integration, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2005, pp. 1–5. doi:10.4230/DagSemProc.04391.22.
[51] P. Riva, M. Žumer, T. Aalberg, LRMoo, a high-level model in an object-oriented framework, 2022.
[52] M. Doerr, M. Theodoridou, CRMdig: A generic digital provenance model for scientific observation, 2011.
[53] D. Soergel, The Art and Architecture Thesaurus (AAT): A Critical Appraisal, 1995. URL: https://www.tandfonline.com/doi/abs/10.1080/01973762.1995.9658306, accessed December 11, 2025.
[54] R. Balzani, S. Barzaghi, G. Bitelli, F. Bonifazi, A. Bordignon, L. Cipriani, S. Colitti, F. Collina, M. Daquino, F. Fabbri, B. Fanini, F. Fantini, D. Ferdani, G. Fiorini, E. Formia, A. Forte, F. Giacomini, V. A. Girelli, B. Gualandi, I. Heibi, A. Iannucci, R. Manganelli Del Fà, A. Massari, A. Moretti, S. Peroni, S. Pescarin, G. Renda, D. Ronchi, M. Sullini, M. A. Tini, F. Tomasi, L. Travaglini, L. Vittuari, Saving

temporary exhibitions in virtual environments: The Digital Renaissance of Ulisse Aldrovandi – Acquisition and digitisation of cultural heritage objects, Digital Applications in Archaeology and Cultural Heritage 32 (2024) e00309. doi:10.1016/j.daach.2023.e00309.

[55] S. Barzaghi, A. Moretti, I. Heibi, S. Peroni, CHAD-KG: A Knowledge Graph for Representing Cultural Heritage Objects and Digitisation Paradata, 2025. doi:10.48550/arXiv.2505.13276.

[56] A. Moretti, S. Barzaghi, CHAD-KG: TTL Serialised RDF Dataset of Exhibited Objects and Digitisation Process, 2025. doi:10.5281/zenodo.16879033.

[57] S. Barzaghi, Simple PUblisher of Knowledge (SPUK), 2025. doi:10.5281/zenodo.15389756.

[58] A. Moretti, S. Barzaghi, CHAD-ASK – LimeSurvey Questionnaire, 2025. doi:10.5281/zenodo.18548589.

[59] P. Ferri, I. Heibi, L. Pareschi, S. Peroni, MITAO: a User Friendly and Modular Software for Topic Modelling, puntOorg International Journal 5 (2020) 135–149. doi:10.19245/25.05.pij.5.2.3.