HECATE at the Crossroads: Guiding Astrophysics into the Semantic Web

Sophia Sideri[1], Iosif Oikonomakis[1], Petros Tsirmakos[2], Giorgos Flouris[1], Vasilis Efthymiou[1,2], Dimitrios Rompogiannakis[3,4], Elias Kyritsis[3,4], Ioanna Leonidaki[3,4], and Andreas Zezas[3,4]

[1] Institute of Computer Science, FORTH, Heraklion, Greece
{sophisid,sifisoik,fgeo}@ics.forth.gr
[2] Department of Informatics and Telematics, Harokopio University, Athens, Greece
{it21693,vefthym}@hua.gr
[3] Physics Department, University of Crete, Greece
ph4502@edu.physics.uoc.gr, {ekyritsis,ioanna,azezas}@physics.uoc.gr
[4] Institute of Astrophysics, FORTH, Heraklion, Greece

Abstract. Massive volumes of raw astrophysical data are collected every day, and subsequently processed into public catalogues used for scientific research. While deep learning has accelerated astrophysics in recent years, the field has yet to benefit from the Semantic Web's advances in data representation and management. As a result, most of those catalogues are typically stored as flat CSV/ASCII files in a tabular format, limiting their interoperability and reuse. In this work, we introduce a semantically enriched astrophysical catalogue, in the form of an ontology-backed Knowledge Graph (KG). Our KG is based on a custom, lightweight ontology (DOCBO), and was populated by transforming the Heraklion Extragalactic Catalogue (HECATE) into RDF using two alternative mapping approaches, X3ML and RML, whose characteristics and trade-offs are discussed and compared. Both the mapping rules and the resulting Knowledge Graphs are openly available to foster reuse, interoperability and community feedback. This work, which is part of the interdisciplinary PARSEC project, constitutes a first proof-of-concept effort towards a more generic methodology for representing astrophysical knowledge of various subdomains (e.g., Supernova Remnants and transients) using semantic technologies, thereby forming and automatically populating semantically-enriched KGs based on appropriately defined ontologies.

Keywords: Knowledge Graph Construction, Astrophysical Data, Data Mappings

1 Introduction

Telescopes, artificial satellites and other astrophysical instruments constantly probe the sky, gathering astronomical data and resulting to large-scale multi-wavelength surveys. The resulting data are pre-processed and organised in *astrophysical catalogues* [20], [24], [30], which contain information about celestial sources such as galaxies, stars and transient events.

Astrophysical research crucially relies on the availability of such catalogues to enable cross-domain research tasks [6]. However, the exponentially growing volume of astronomical data highlights the need for a new approach in astronomical catalogue curation, as these surveys (and the resulting catalogues) constitute enormous datasets that contain valuable information, but often lack the semantic structure needed to fully support reuse, integration, and automated exploration appropriate for downstream tasks [1].

In practice, catalogue curation remains a manual and time consuming task. As the volume of incoming data increases, along with the different data formats, the task of catalogue curation poses major challenges. Catalogues are currently stored as static tables in CSV or other textual formats, or, in the best scenario, as a single-table relational database [12]. As a result, these resources risk becoming outdated, difficult to maintain, and hard to exploit in new research contexts [27].

Semantic technologies and Knowledge Graphs (KGs) [2], [13] could provide a solution to this problem, by representing the data as entities and relationships, aligned with a domain-specific ontology that defines concepts and their properties. This model could be used to perform data querying in more expressive and semantically meaningful ways, and can allow efficient data searching, querying, reasoning, data analysis, pattern identification, provenance, trust and credibility analyses, catalogue maintenance and enrichment, data reuse, catalogue interlinking, and others. Nowadays, the astrophysical community have introduced ontologies for astronomical objects and metadata [17], [16], but large-scale catalogue conversion into RDF-based KGs remains rare, due to the scale, complexity, and flatness of the existing datasets.

The development and application of a machine-assisted methodology for constructing and curating semantically enriched astrophysical catalogues in the form of KGs is the main objective of the PARSEC project[1]. In this paper, we focus on the subproblem of automatically constructing a domain-specific KG based on the data contained in the Heraklion Extragalactic Catalogue (HECATE) [19], an astrophysics catalogue which contains information about extragalactic sources. We present a methodology for transforming this catalogue into a semantic KG, called **HECAT3KG**, using a custom lightweight ontology named **DOCBO**, tailored to the domain. For the population, we explored different mapping frameworks, in particular the RML [8] and X3ML [21] mapping languages.

To enhance reproducibility, the resulting resources are publicly available on Zenodo[2], whereas the code and mappings are available on Github[3]. All resources are released under open licenses (CC BY 4.0 for data and Apache-2.0 for code).

The rest of the paper is organized as follows. In Section 2, we review related work on astrophysical catalogues and semantic mapping frameworks. Section 3 describes the ontology, some competency queries used for its validation, and the

[1] Propelling Astrophysics Research via Semantically Enriched Catalogues - `https://isl.ics.forth.gr/PARSEC/`.

[2] DOCBO: `https://zenodo.org/records/16281112`, HECAT3KG: `https://doi.org/10.5281/zenodo.15379419`

[3] `https://github.com/sophisid/HECAT3KG`

mappings used for its population. Section 4 provides a comparison among the two mapping approaches that we used. Finally, Section 5 concludes the paper and outlines directions for future work.

2 Background and Related Work

In this section, we provide some background knowledge regarding HECATE, the astrophysics catalogue that we use throughout this paper, and then describe and compare against existing efforts for KGs in astrophysics, as well as mapping languages for transforming (semi-)structured data into RDF.

2.1 HECATE

The Heraklion Extragalactic Catalogue[4] (HECATE) [19] describes 204,733 galaxies in the local Universe (i.e., with distance less than 200 Mpc). Galaxies are individually selected from the HyperLEDA astronomical database, mainly based on radial velocity ($< 14,000$). For this reason, the methodology adopted for constructing the catalogue is specifically tailored to increase the quality of the data for nearby galaxies and to facilitate cross-matching procedures.

The main features of HECATE are the calculation of robust distances based on the combination of available redshift-independent indicators and redshifts, along with homogenized size information, and the computation and homogenization of star-formation rates based on infrared indicators.

HECATE provides various characteristics for each of the featured galaxies, e.g., position, morphological information, distance, star-formation rate, stellar mass, gas-phase metallicity, nuclear classification and others. The wealth of this information can be particularly useful for a wide-range of studies such as: the exploration of nearby galaxies and their properties; the selection of galaxy samples for future surveys; the identification of the host galaxies of multi-wavelength source populations and the characterization of their connection with their host's parameters; and the application of astrophysical priors in the search for electromagnetic counterparts of gravitational-wave sources.

As mentioned, HECATE is essentially a large table, each line of which describes one galaxy. The columns of this table contain measurements or other information about different physical properties of the respective galaxy. From an ontological representation perspective, HECATE contains four kinds of columns:

1. *Identification columns*, which contain galaxy identification information taken from other catalogues, thereby helping determining the galaxy that the respective line in the catalogue corresponds to.
2. *Measurement columns*, which contain the measurements regarding the galaxies' attributes and physical properties.

[4] `https://hecate.ia.forth.gr/`

3. *Error and confidence interval columns* about (some of) the measurements that appear in the measurement columns; these are very important for the astrophysicists and are related to the accuracy of the instruments and methodologies used to determine a measurement.
4. *Flag columns*, which provide metadata about other data columns, such as the reliability and context of measurements (e.g., a flag could indicate whether a measurement is reliable or not).

2.2 KGs for Astrophysics

Creating and maintaining astrophysical catalogues is a long-standing problem for astrophysics research, currently being accomplished mainly by manual effort, either by catalogue curators who detect and ingest the data, as in the case of NASA/IPAC Extragalactic Database (NED) [24] and HyperLEDA [20], or by researchers and scientific journals that notify the catalogues of interest about their new findings, as in the case of SIMBAD [30]. Some astrophysics studies of wide impact (e.g., [11]) were, in essence, the manual work of several months, combining astrophysics literature and catalogues.

However, such catalogues are usually in semi-structured, flat tabular formats, expressed in plain ASCII, CSV or MRT files (the latter are similar to CSV). The transformation of existing scientific findings into a machine-readable and reusable format, as recommended by the linked data principles [5], will improve the catalogues' expressiveness, facilitate their reuse, maintenance and update, and allow added-value services to be implemented on top of them, such as catalogue integration and interlinking, expressive provenance recording, etc. As a result, this transition would significantly raise the value of catalogues and help scientists in their data-intensive research.

Efforts to address these issues through ontologies and KGs have been proposed. For example, ASON [17], models astrophysical services and objects, emphasizing service interoperability. Similarly, the International Virtual Observatory Alliance (IVOA) standards [16] provide vocabularies for celestial objects and metadata, enabling semantic integration across datasets. However, IVOA's ontology maintenance has been discontinued, limiting its applicability for ongoing projects. These approaches, while valuable, tend to be overly expressive and complex, making them difficult to apply efficiently on flat and large-scale datasets such as HECATE. In our case, such ontologies would significantly increase the number of generated triples, which could hinder scalability in future use cases, especially in the context of streaming data.

2.3 Mapping Languages

Transforming semi-structured data into RDF is a fundamental step for creating a KG. Mapping languages allow the expression of rules that determine how to represent the source data as RDF triples, in accordance to the target ontology. Over the past years, many mapping frameworks have been developed to translate the data from formats like CSV, XML or relational databases into semantic

representations. Among the most used tools are X3ML [21] and RML [8], each of which serves specific purposes and has been used in various projects requiring RDF materialization.

X3ML [21] is a declarative, XML-based language designed for clear, readable mappings. It was originally created to support mappings in the cultural heritage domain [22], where the complexity of the data often requires a more flexible and understandable set of transformation rules. X3ML has been used in various projects on diverse domains [4], [10], [26], in environments relying on semantic interoperability for connecting and reusing data among different disciplines. One of the key strengths of X3ML is its focus on human-readable mappings that can be written, reviewed, and maintained by both technical and domain experts. The X3ML framework has also been extended with tools like instance and label generator, and is supported by the 3M Mapping Memory Manager [18] interface, which simplifies collaboration between engineers and domain experts, like astrophysicists.

The *RDF Mapping Language (RML)* [8] is widely used, as it is a superset of the standard knowledge graph construction engine, R2RML [25], which is a W3C recommendation. While R2RML is restricted to relational databases and relies on SQL queries, RML has been extended to support additional data formats such as CSV, JSON and XML, offering flexibility while retaining the expressivity of the mapping syntax. However, RML mappings can become quite complex and are often difficult for non-technical users to understand and manage. To address RML's complexity and performance, recent works such as the modular redesign of the RML ontology [14] and the development of scalable processors like Morph-KGC [3] and SDM-RDFizer [8] have been proposed. These efforts aim to improve expressiveness, performance and scalability, particularly for large heterogeneous datasets. Comparative studies of mapping frameworks such as RML, R2RML, and YARRRML [7], [15], [29] have shown that RML excels in automation and scalability.

3 Designing and Populating the Ontology

3.1 Ontology Design and Motivation

Although standard ontologies to represent scientific measurements and observational data exist, they tend to be complex, overly general, or not suited for the scale and flat structure of HECATE. To address this, we developed a domain-specific ontology, DOCBO (Domain Ontology for Celestial Bodies and Observations), having in mind the structure and semantics of HECATE.

The DOCBO ontology (see Figure 1) provides a lightweight but expressive model to describe astrophysical records expressed in tabular formats, i.e., the format of most astrophysical data. It was developed using Protégé [23], and contains 15 classes, 18 object properties and 15 data properties. As shown in Figure 1, the ontology models:

- Celestial bodies (`E02 Celestial Body`), namely objects in the universe (galaxies in the case of HECATE). Note that celestial bodies can contain

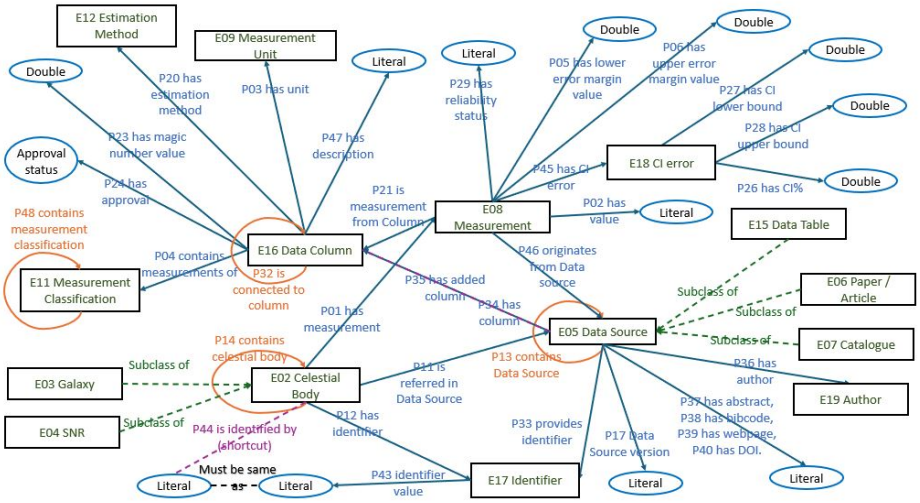

Fig. 1: The DOCBO ontology.

other celestial bodies (a galaxy can contain a star system, a star system can contain a planet, etc).
- Identifiers (`E17 Identifier`), which can be used to identify and disambiguate celestial bodies that often have different identifiers in different data sources.
- Data sources (`E05 Data Source`), which are used to record provenance information, i.e., the data table, article, or catalogue from which the information was derived.
- Measurements (`E08 Measurement`), which are used to record the actual values (measurements) of different physical quantities found in the table. Various metadata on the measurements can be recorded (such as error margins, measurement reliability, measurement unit, estimation method etc), some of which are recorded using intermediate classes for proper modelling.

These entities capture the semantics of HECATE entries while allowing future ontology alignment with other relevant ontologies in future work. A more detailed documentation of these classes and properties is provided in the accompanying Zenodo resource, found in `https://zenodo.org/records/16281112`.

3.2 Competency Queries

To assess the usefulness of DOCBO, a number of competency queries were developed, capturing various features and types of searches that astrophysicists typically perform. All these queries are supported by the current version of DOCBO. Due to space considerations, only a partial list of these competency queries is included here, expressed in natural language.

1. Return all available photometric data and magnitudes for a given galaxy.

2. Give all sources with ages > 103yrs or diameter > 50pc or distance < 3Mpc.
3. For a given galaxy, how many/which publications present shock velocity/proper motion measurements and which are those values?
4. Return all galaxies with sizes < 0.5 degrees from SDSS and distances between 100 and 200 Mpc.
5. Return all galaxies with photometry g < 20.0 mag, SNR > 3.0 and activity classification AGN, based on spectroscopy.
6. Return all pairs of galaxies having distances less than 10 arcsec
7. Return all systems of galaxies that contain three objects or more, have distances less than 10 arcsec, and g-band photometry that differs more than 1.0 mag at the 90% confidence.

3.3 Mappings for Ontology Population

To populate the HECAT3KG Knowledge Graph with instances from HECATE (which contains more than 200,000 records and each more than 100 columns stored in a single CSV), we examined two different mapping frameworks: X3ML [21] and RML [8]. Both approaches can be used to transform the source data into RDF triples, based on the target ontology described above, DOCBO. Each mapping approach offers different strengths and trade-offs in terms of readability, maintainability, and performance, and we compare their effectiveness in handling large-scale astrophysical data (see Section 4). Interestingly, both mapping approaches required a preprocessing step to split the initial dataset into several ones that are processed individually. Below, we present the two approaches, describing the mapping workflow, preprocessing steps, and their main strengths and limitations.

X3ML Mappings For the X3ML mappings, we adopted a declarative approach which helped collaboration with the domain experts. X3ML divides the mapping process into two parts: (i) the *Mapping Definition Document*, which defines how source data elements translate to ontology concepts and properties, and (ii) the *Generator Policy Document*, which defines templates on how new Unique Resource Identifiers (URIs) and literals are going to be generated. The transformation was performed using the X3ML Engine[5].

Our initial approach was to create individual mapping rules, and to transform the values for each column in the dataset; this is very much like the way a domain expert would approach the data for the first time. To support this, we transformed the CSV into an XML format, where each row became a `<ROW/>` element, and each column was a nested element named after its column header (e.g., `<RA/>`, `<DEC/>`, for right ascension and declination, respectively). We then treated each XML element and attribute independently, resulting in one-to-one mappings. This approach is simple and straightforward, but also computationally inefficient. With more than 200,000 entities and over 100 rules, the mapping engine encountered performance bottlenecks.

[5] `https://www.ics.forth.gr/isl/x3ml-toolkit`

To tackle the above issues, we implemented a second approach that introduces a preprocessing step that transformed the CSV into a semantically organized XML structure. This structure followed a schema that reflected the domain semantics and exhibited mapping efficiency. For instance, identifiers were grouped under the same XML node `<identifiers>`, positional data under `<degrees>`, orientation data in `<rotation>`, measurement uncertainties in `<errors>` and confidence intervals under `<confidence_intervals>`. Flags and classifications were included as standalone elements, except from flux data from IRAS[6] and distance-related flags, which were placed in `<flux>` and `<DIST>` respectively. The remaining column categories were placed in a generic block, named `<gen>`.

The key benefits of X3ML are that it uses an intuitive, human-readable structure, making it easy for both engineers and astrophysicists to review and validate the mappings. Additionally, X3ML allows tracking how the RDF data was produced, by the mapping rules that produced it. On the other hand, X3ML is XML-based, which limits its applicability when working with different data source formats, requiring a preprocessing step. This also causes efficiency issues when dealing with many flat data instances, as was the case with HECATE.

RML Mappings For RML, we utilized the RMLMapper[7] engine to transform the HECATE CSV into RDF, leveraging RML's native support for CSV, JSON, XML and relational databases. RML defines declarative mappings as Turtle files, specifying how CSV columns correspond to ontology classes and properties. However, the flat structure of HECATE CSV and RML's lack of native conditional logic for CSV inputs, necessitated a preprocessing step to ensure semantic accuracy and minimize redundant triples.

The preprocessing enriches the CSV by adding 10 auxiliary columns to support mappings. These include error margins, reliability status, flux quality descriptions from IRAS flux flags, metallicity status, source labels, and distance estimation methods. For example, **RFLAG** values are mapped to descriptions like "Reliable"/"Unreliable" (for the values 1 or 0 respectively). Additionally, the **F_ASTROM** field, which indicates astrometric uncertainty of the galaxy's position, is used to compute an error margin using the formula $2.78 \times 10^{-4+F_ASTROM}$. However, the preprocessing does not filter rows with invalid measurements, resulting to redundant triples. While this increased the CSV size, it facilitated precise RDF triple generation.

To manage the mapping for the ∼100 columns, we developed a tool to automate rule generation for measurement columns. In a subsequent step, we refined the URI generation using custom functions to ensure consistency and semantic clarity. The resulting URIs follow a clear and reusable naming template like: `https://parsec.forth.gr/celestial_body/galaxy_{PGC}/column/{column}/measurement/...` . In this template, `PGC` refers to a Principal Galaxy Catalogue (PGC) Identifier, which is a unique numerical ID assigned to each galaxy, while `column` represents the name of a specific column in the dataset

[6] `https://lambda.gsfc.nasa.gov/product/iras/`
[7] `https://github.com/RMLio/rmlmapper-java`

(such as `r1`, `metal`, etc). These placeholder values are dynamically replaced with the actual ones during the URI generation process.

Another challenge is RML's limited ability to filter empty or invalid values, leading to redundant RDF triples for cases like missing measurements. However, in some cases, it was not possible to identify or label all invalid values during preprocessing — either because the condition was too complex to encode, or due to missing metadata in the source. This results in a larger KG, but it does not affect the semantics of the output KG, or the results of the queries that we want to perform over it. Additionally, RML required around 100 mapping rules, i.e., one mapping per column, increasing the maintenance complexity.

On the other hand, RML excelled in performance using the RMLMapper, even when producing the same or bigger KG compared to X3ML. Its versatility to handle diverse data formats makes it suitable for future integration. However, it tends to be more technical.

3.4 Developed Resources

Both the DOCBO ontology and the two KGs (HECAT3KG-X3ML and HECAT3KG-RML, which were derived from the HECATE dataset using the X3ML and RML mappings respectively), adhere to the FAIR principles [31]. The ontology and the produced KGs are *Findable* through their publication on Zenodo with persistent identifiers (DOIs). They are *Accessible* on Zenodo and Github. The KGs are *Interoperable*, as they are linked to external datasets (astrophysical catalogues) through their identifiers. The use of RDF and a custom ontology ensures *Reusability* by providing a machine-readable, semantically rich data.

4 Comparison of Mapping Approaches

As mentioned, the two mapping approaches we used generated different KGs, HECAT3KG-X3ML and HECAT3KG-RML; this was due to the distinct characteristics of the X3ML and RML mapping frameworks. However, it should be noted that both KGs preserve the semantics of the HECATE dataset, mapping the records of all galaxies to DOCBO classes and properties. In other words, the aforementioned variations do not affect the correctness of query results, but they do affect the storage and maintenance process.

The characteristics of these two KGs and of the mapping approaches employed are summarized in Tables 1 and 2, and explained in more details below. It is important to note that our comparison should not be used to infer generic conclusions about the two approaches. Indeed, our work is not meant to be a general comparison of the two mapping languages (RML and X3ML), as such a comparison would require a more thorough theoretical and experimental setting that would take full advantage of their capabilities. Instead, in this paper we only report on our experiences and findings for our particular use case and our implementation of those approaches.

Table 1: X3ML Engine vs RML Mapper: Performance Metrics

Metric	**X3ML Engine**	**RML Mapper**
Execution Time	∼18 hours	∼50 minutes
Output Size	∼5 GB	∼9 GB
Number of Triples	∼44 Million	∼55 Million

Execution time. In our original application of the mapping rules over HECATE, we experienced a very slow execution time. Upon further tests, we realized that the problem was related to the fact that the input file with all the galaxy entities was too large to be processed efficiently. The problem appeared in both mapping languages, but was much more pronounced for the X3ML language.

To address this problem, we split the original input file into smaller chunks that were processed independently. For X3ML, we split the original file into smaller chunks of 100 galaxies each, resulting to a total of $\sim 2,000$ smaller files. As RML was fairly efficient, we used a more coarse-grained splitting, resulting to ∼200 files containing 1,000 galaxies each. In both cases, after the migration of each smaller chunk in an RDF file, all the output files were merged and uploaded into the target triple store, resulting to the expected RDF graph.

The times reported in Table 1 were measured under the above process. As seen in the table, X3ML ran in ∼18 hours, whereas RML in just ∼50 minutes. This was despite the fact that X3ML uses less rules and produces a smaller KG (see Table 1). The main reason for this finding is that RML mappings are processed using engines optimised for speed, like RMLMapper [9], leading to low execution times. On the other hand, X3ML is designed with other goals in mind, namely good human-readability and easiness of collaboration, rather than execution speed. Moreover, for our use case, X3ML required the intermediate CSV to XML transformation, which introduced an additional overhead.

These results highlight performance trade-offs. Future applications on larger or even streaming astrophysical catalogues may introduce further scalability challenges. Utilizing incremental or distributed processing of the transformation could help alleviate this problem.

Size of resulting KGs. HECAT3KG-X3ML and HECAT3KG-RML differ vastly with respect to their size. As Table 1 shows, X3ML produced a much smaller file

Table 2: X3ML vs RML Language: Features and Characteristics

Feature	**X3ML**	**RML**
Input Formats	XML	CSV, XML, JSON, RDBs
Mapping Rules	10	157
Mapping Intuitiveness	High	Low
Column Merging	Yes	With preprocessing
Conditionals	Yes	Partly, with preprocessing
URI Consistency	Yes	With preprocessing

(with size of about 5GB), containing 44 Million triples; on the other hand, RML produced a file size of about 9GB, with 55 Million triples. The main reason for this was the conditional limitations in RML with CSV files (see below), particularly for the uncertainties (errors) in measurements and the related flags, which caused RML to produce more triples than needed, and eventually resulting to a much bigger output file.

Input formats. An important difference between the two approaches is that they support different input formats (see Table 2). X3ML requires XML data, which raises the need for a preprocessing step whenever the data are in another format (as in our use case), whereas RML supports a variety of input types, including CSVs, eliminating the need for format conversion. We opted not to transform the CSV into XML for the RML approach, as this would undermine its design for handling different formats.

Intuitiveness and size of the mappings. When creating the mapping rules, X3ML was found to be more intuitive and user-friendly, especially for domain experts without technical knowledge or background. Using the 3M Editor [28], as well as the instance and label generators, astrophysicists were able to actively participate in the mapping process. By leveraging the semantic structure we introduced in the preprocessing step, we managed to reduce the X3ML mapping rules to 10, which increases the intuitiveness and simplifies the maintenance of the mappings. For comparison, RML required the definition of a separate rule for nearly every column, leading to 157 mappings that are more complex and difficult to maintain (see Table 2). Moreover, the syntax of RML is more technical and less accessible for non-experts.

Column merging. With regards to data mapping functionalities, X3ML leverages XPath, thus allowing the creation of data triples from information found in two or more different columns without the need for any preprocesing. This proved very handy during the creation process for the mapping rules. On the other hand, RML does not support this functionality in CSVs; to achieve the same effect, an additional temporary column had to be created (containing the combined information) and used for the mapping (see Table 2).

Conditionals. X3ML's XPath conditional logic helps skip empty or invalid values, producing fewer redundant triples. In contrast, RML resulted in more triples for missing values in the CSV, like measurement flags and error margins (see Table 2). The preprocessing step tried to mitigate this by labeling the invalid values, but the main issue persisted, leading to the size difference between HECAT3KG-RML and HECAT3KG-X3ML that is shown in Table 1.

URI consistency. Another important aspect for our use case was URI consistency, which refers to the ability of the mapping language to generate a uniform and intuitive set of URIs whose naming makes sense for a human reader and can be used to track the origin of the information in the source data. We should

note here the importance of URI consistency with respect to the pipeline we followed for the processing of HECATE: the splitting of the original XML file into many, and the subsequent merging of the resulting outputs to a single RDF file, would not produce a correct result without this feature, as cross-referencing among different chunks would fail, resulting to wrong mappings. URI consistency is supported natively by X3ML through its URI generator policies, ensuring consistent and traceable URIs even when data is processed incrementally. On the contrary, RML does not support URI generation policies, so we had to manually create appropriate URI naming schemes during the preprocessing step to avoid inconsistencies in the execution of our pipeline (see Table 2).

Summary of findings. In summary, the two approaches had significant trade-offs related to their semantics, performance and usability for the described use case. RML excels performance-wise and can handle different data formats, making it suitable for heavy integration scenarios. On the other hand, X3ML can create rich and expressive semantics while allowing the mapping process to be more intuitive and more easily graspable by domain experts.

5 Conclusions

In this paper, we described our methodology for transforming the HECATE astrophysical catalogue into a semantic Knowledge Graph (KG), HECAT3KG, using a custom ontology (DOCBO) and two mapping frameworks (X3ML and RML). In astrophysics, large amounts of data are generated constantly, and stored in a tabular form using unstructured or semi-structured formats (ASCII, CSV) in large text files (called catalogues). The aim of this work, which is part of the more ambitious PARSEC project, was to provide a proof-of-concept methodology to migrate one of those catalogues, namely HECATE [19], into a KG. Our more general objective is to show how structured formats, like KGs, can be used to enhance the value of astrophysical catalogues by allowing the development of added-value services (e.g., related to maintenance and curation, provenance and trust management, advanced querying and navigation, etc).

Our contributions include: (i) the DOCBO ontology tailored to celestial objects and measurements, (ii) the HECAT3KG knowledge graph constructed using X3ML and RML, and (iii) comparative insights into the mapping frameworks' usability and performance.

The generated resources are publicly available in Zenodo[8] and Github[9] under open licenses. While X3ML facilitated collaboration with domain experts, RML offered better performance and scalability. Our methodology has already been reused in an ongoing effort to build a KG for Supernova Remnants (SNRs), showing its applicability to other domains.

[8] `https://doi.org/10.5281/zenodo.15379419` and `https://zenodo.org/records/16281112`

[9] `https://github.com/sophisid/HECAT3KG`

For future work, we plan to align DOCBO with existing standards (such as CIDOC CRM extensions) and other astrophysical ontologies and vocabularies (e.g., DCAT, Measurement Ontology, IVOA), defining a long-term maintenance and update strategy for the ontology and knowledge graph.

Acknowledgments. This work was partially funded by the "Theodore Papazoglou" FORTH Synergy Grants, through the PARSEC project. We would like to thank David Chaves-Fraga for his valuable guidance in R2RML/RML tools.

References

1. Accomazzi, A., Eichhorn, G., Kurtz, M.J., Grant, C.S., Murray, S.S.: The nasa astrophysics data system: Architecture. Astronomy and Astrophysics Supplement Series **143**(1), 85–109 (Apr 2000)
2. Antoniou, G., Groth, P., Harmelen, F.V., Hoekstra, R.: A Semantic Web Primer. MIT Press, 3rd edn. (2012)
3. Arenas-Guerrero, J., Alobaid, A., Navas-Loro, M., Pérez, M.S., Corcho, Ó.: Boosting knowledge graph generation from tabular data with RML views. In: ESWC-23. LNCS, vol. 13870, pp. 484–501. Springer (2023)
4. Basset, A., Los, W.: Biodiversity e-science: Lifewatch, the european infrastructure on biodiversity and ecosystem research. Plant Biosystems-An International Journal Dealing with all Aspects of Plant Biology **146**(4), 780–782 (2012)
5. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. In: Seneviratne, O., Hendler, J.A. (eds.) Linking the World's Information - Essays on Tim Berners-Lee's Invention of the World Wide Web, ACM Books, vol. 52, pp. 115–143 (2023)
6. Budavári, T., Szalay, A.S.: Probabilistic cross-identification of astronomical sources. The Astrophysical Journal **679**(1), 301–309 (May 2008)
7. Dimou, A.: R2RML and RML comparison for RDF generation, their rules validation and inconsistency resolution. CoRR **abs/2005.06293** (2020)
8. Dimou, A., Sande, M.V., Colpaert, P., Verborgh, R., Mannens, E., de Walle, R.V.: RML: A generic language for integrated RDF mappings of heterogeneous data. In: Proceedings of the Workshop on Linked Data on the Web. vol. 1184 (2014)
9. Dimou, A., Vander Sande, M., Colpaert, P., Verborgh, R., Mannens, E., Van de Walle, R.: Rml: A generic language for integrated rdf mappings of heterogeneous data. LDOW **1184** (2014)
10. Doerr, M., Theodoridou, M.: Mapping archaeological databases to CIDOC-CRM. In: Proceedings of the 40th Annual Conference on Computer Applications and Quantitative Methods in Archaeology (CAA) (2011)
11. Green, D.: SNR catalogue (online resource). `https://www.mrao.cam.ac.uk/surveys/snrs/snrs.paper.html`, accessed: April 30, 2025
12. Hanisch, R.: Data standards for the international virtual observatory. Data Science Journal **5**, 168–173 (2006)
13. Hitzler, P., Krötzsch, M., Rudolph, S.: Foundations of Semantic Web Technologies. Chapman & Hall/CRC (2009)
14. Iglesias-Molina, A., Assche, D.V., Arenas-Guerrero, J., Meester, B.D., Debruyne, C., Jozashoori, S., Maria, P., Michel, F., Chaves-Fraga, D., Dimou, A.: The RML ontology: A community-driven modular redesign after a decade of experience in mapping heterogeneous data to RDF. In: ISWC-23. vol. 14266, pp. 152–175 (2023)

15. Iglesias-Molina, A., Chaves-Fraga, D., Dasoulas, I., Dimou, A.: Human-friendly RDF graph construction: Which one do you choose? In: ICWE-23. vol. 13893, pp. 262–277. Springer (2023)
16. IVOA: IVOA vocabulary. https://www.ivoa.net/rdf/object-type/2020-10-06/object-type.html, accessed: April 30, 2025
17. Karray, F., Ahmed, M.B., Kacem, A.H.: ASON: An OWL-S based ontology for astrophysical services. Computers & Geosciences **94**, 71–80 (2016)
18. Konsolaki, K., Samaritakis, G.: 3m: Mapping memory manager. https://www.ics.forth.gr/3m-mapping-memory-manager (2015), accessed: 2025-04-26
19. Kovlakas, K., Zezas, A., Andrews, J.J., Basu-Zych, A., Fragos, T., Hornschemeier, A., Kouroumpatzakis, K., Lehmer, B., Ptak, A.: The Heraklion Extragalactic Catalogue (HECATE): a value-added galaxy catalogue for multimessenger astrophysics. Monthly Notices of the Royal Astronomical Society **506**(2), 1896–1915 (2021)
20. Makarov, D., Prugniel, P., Terekhova, N., Courtois, H., Vauglin, I.: HyperLEDA. III. the catalogue of extragalactic distances. A&A **570**, A13 (oct 2014)
21. Marketakis, Y., Minadakis, N., Samaritakis, G., Doerr, M.: X3ML mapping framework for information integration in cultural heritage and beyond. International Journal on Digital Libraries **18**(4), 301–319 (2017)
22. Minadakis, N., Marketakis, Y., Kondylakis, H., Flouris, G., Theodoridou, M., de Jong, G., Doerr, M.: X3ML framework: An effective suite for supporting data mappings. In: Ronzino, P. (ed.) Proceedings of the Workshop on Extending, Mapping and Focusing the CRM. CEUR Workshop Proceedings, vol. 1656, pp. 1–12 (2015)
23. Musen, M.A., Protégé Team: The protégé project: A look back and a look forward. AI Matters **1**(4), 4–12 (Jun 2015)
24. NASA/IPAC: NASA/IPAC extragalactic database (ned). http://ned.ipac.caltech.edu, accessed: April 30, 2025
25. Pinkel, C., Binnig, C., Haase, P., Martin, C., Sengupta, K., Trame, J.: How to best find a partner? An evaluation of editing approaches to construct R2RML mappings. In: ESWC-14. LNCS, vol. 8465, pp. 675–690. Springer (2014)
26. Remy, L., Ivanovic, D., Theodoridou, M., Kritsotaki, A., Martin, P., Bailo, D., Sbarra, M., Zhao, Z., Jeffery, K.G.: Building an integrated enhanced virtual research environment metadata catalogue. Electron. Libr. **37**(6), 929–951 (2019)
27. Richards, J.W., Starr, D.L., Butler, N.R., Bloom, J.S., Brewer, J.M., Crellin-Quick, A., Higgins, J., Kennedy, R., Rischard, M.: On machine-learned classification of variable stars with sparse and noisy time-series data. The Astrophysical Journal **733**(1) (apr 2011)
28. Samaritakis, G., Konsolaki, K.: 3meditor: Web application for editing/viewing x3ml mapping files (2019), https://github.com/isl/3MEditor, accessed: 2025-04-26
29. Warren, P., Mulholland, P., Daga, E., Asprino, L.: Path-based and triplification approaches to mapping data into rdf: User behaviours and recommendations. Semantic Web **15**(6), 2479–2505 (2024)
30. Wenger, M., Ochsenbein, F., Egret, D., Dubois, P., Bonnarel, F., Borde, S., Genova, F., Jasniewicz, G., Laloë, S., Lesteven, S., Monier, R.: The SIMBAD astronomical database. the CDS reference database for astronomical objects. Astron. Astrophys. Suppl. Ser. **143**, 9–22 (Apr 2000)
31. Wilkinson, M.D., Dumontier, M., Aalbersberg, I.J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.W., da Silva Santos, L.B., Bourne, P.E., et al.: The fair guiding principles for scientific data management and stewardship. Scientific data **3**(1), 1–9 (2016)