

Building Wikidata-Enriched Applications through Composable Semantic Transformation Pipelines with Chimera

Marco Grassi^{*,†}, Mario Scrocca^{*,†}, Alessio Carenini and Irene Celino

Cefriel – Politecnico di Milano, Milan, Italy

Abstract

Wikidata is increasingly used as a shared knowledge hub for enriching application-specific datasets, yet integrating it with heterogeneous data sources still requires addressing differences in formats, schemas, access mechanisms, and target application requirements. This demo paper presents the use of Chimera, an open-source framework for building composable semantic transformation pipelines, to support data harmonisation and fusion with Wikidata. The demo focuses on a mobility scenario in which public transport stops are retrieved from a GTFS Schedule feed, lifted into RDF according to a reference ontology, enriched with nearby landmarks from Wikidata via dynamically generated queries, and finally converted to the target format required by a dashboard application. Chimera supports this workflow by combining the execution of knowledge conversion mapping rules, RDF graph operations, and Apache Camel integration patterns into configurable pipelines. The demo shows how Chimera reduces the effort required to implement data harmonisation and integration workflows leveraging the knowledge represented within the Wikidata graph.

Keywords

Semantic Conversion, Data Pipeline, Knowledge Graph Construction, Mapping Languages

1. Introduction

Wikidata provides a large, openly accessible knowledge graph [1] that can be queried and reused in a wide range of applications, from data enrichment to exploratory analytics and semantic search. Its value, however, is not limited to direct querying: Wikidata can also act as an integration layer that connects local, domain-specific datasets with shared identifiers, descriptions, classifications, and relationships. This is particularly relevant in application scenarios where organisations need to combine internal or domain-specific data with external knowledge while keeping the resulting workflow maintainable and reproducible. At the same time, the practical integration of Wikidata with heterogeneous data sources remains challenging.

Application data often comes in non-RDF formats such as CSV, JSON, XML, ZIP archives, or API responses, while target systems may require application-specific output formats. Moreover, integration workflows typically involve multiple steps: accessing data, harmonising it through a reference ontology, enriching it with external knowledge, querying or validating the resulting graph, and exporting it to the expected target representation. Different use cases require flexible combinations of specialised operations, depending on data size, frequency, performance requirements, and integration constraints. As a result, custom dedicated components are usually developed, with issues related to maintainability and reusability.

This demo paper showcases how the Chimera¹ framework could address these challenges by supporting the development of data harmonisation and fusion solutions integrated with Wikidata.

Wikidata'26: Wikidata Workshop at ISWC 2026

*Corresponding author.

[†]These authors contributed equally.

✉ marco.grassi@cefriel.com (M. Grassi); mario.scrocca@cefriel.com (M. Scrocca); alessio.carenini@cefriel.com (A. Carenini); irene.celino@cefriel.com (I. Celino)

ORCID 000-0003-3139-3049 (M. Grassi); 0000-0002-8235-7331 (M. Scrocca); 0000-0003-1948-807X (A. Carenini); 0000-0001-9962-7193 (I. Celino)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

¹<https://github.com/cefriel/chimera>

Chimera is implemented by building on the Apache Camel² integration framework and provides modular components for semantic data transformation pipelines, including RDF graph operations and knowledge conversion operations [2]. It allows developers and data engineers to define pipelines declaratively, combining Chimera components with production-ready Camel components for integration with heterogeneous protocols and services.

The presented solution has been developed for the *Knowledge Graphs for Data Interoperability with Chimera* tutorial³ [3] that provides additional material on Chimera. By focusing on a concrete mobility use case, the goal is to populate an interactive dashboard showing public transport stops and nearby landmarks by processing transport operator feeds and Wikidata entities. In this paper, we present in detail the approach followed with a specific focus on how Wikidata-based enrichment can be operationalised within reusable semantic transformation pipelines. The aim is to make the approach available beyond the tutorial setting and to demonstrate how other applications can leverage Chimera in a similar fashion to integrate local data with Wikidata and produce application-ready outputs.

The remainder of the paper is organised as follows. Section 2 presents related work and background concepts, Section 3 introduces the demo scenario, Section 4 describes its implementation using Chimera, Section 5 discusses its implications for Wikidata-based applications, and Section 6 concludes the paper.

2. Related Work and Preliminaries

Wikidata is increasingly used as a shared knowledge hub to enrich application-specific datasets with identifiers, labels, descriptions, classifications, and relationships that can be retrieved via the Wikidata Query Service [4]. Recent work such as KIF [5], adopts the Wikidata data model and vocabulary as a common abstraction for virtually integrating heterogeneous knowledge sources through user-defined mappings. Our approach materialises source data as RDF and uses Wikidata as one of the enrichment sources, while allowing the resulting graph to conform to an application-specific ontology and to be serialised into the format required by the target system. Consequently, the approach is not tied to Wikidata as the reference model, although it incurs the computational and storage costs associated with RDF materialisation.

Declarative mapping languages are central to constructing knowledge graphs from heterogeneous data using a reference ontology [6]. Many mapping languages and related processors have emerged to provide maintainable and scalable ETL solutions for this task [7]. Mapping rules can be defined and executed to bind input data sources to Wikidata entities. These tools are highly relevant for the lifting phase of a semantic integration workflow, but they primarily focus on generating RDF graphs from source data (*lifting*) rather than orchestrating complete application-oriented pipelines. An *any-to-RDF-to-any* mapping approach can support interoperability and integration across various information systems by enabling a *lowering* procedure from RDF to the target output format [8, 9].

Federated query systems provide another complementary approach [10]. Such frameworks enable querying across multiple SPARQL endpoints and may include Wikidata among the queried sources. This is effective when data is already available as RDF or exposed through SPARQL interfaces.

In the proposed demo, we leverage the Chimera [2] framework, built on top of Apache Camel, to operationalise Wikidata as an enrichment source within an executable and reproducible pipeline for data harmonisation and fusion. Apache Camel provides a large catalogue of reusable components⁴ for interacting with heterogeneous data sources and sinks, including file systems, relational databases, message brokers, and web services. These components can be composed through Enterprise Integration Patterns (EIPs) [11] to define executable data processing pipelines. Building upon this foundation, Chimera extends Apache Camel with domain-specific components tailored to Semantic Web applications. In particular, the graph component enables RDF graph management operations, such as

²<https://camel.apache.org/>

³<https://cefruel.github.io/kg4di/>

⁴<https://camel.apache.org/components/4.18.x/>

querying and updating remote triplestores through SPARQL, while the mapping-template component supports data harmonisation through declarative transformations specified in the Mapping Template Language (MTL)⁵ [9] or RML [12]. A comparison of Chimera against alternative approaches for knowledge graph construction and RDF-based data integration pipelines is provided in [2]. Chimera differs mainly in its integration with Apache Camel, which allows semantic data operations to be combined with a broader catalogue of connectors and EIPs to satisfy heterogeneous integration requirements, but has the drawback of requiring familiarity with the framework and the Java ecosystem.

In this paper, we focus on the materialisation of an integrated graph combining data from structured data sources (i.e., set of CSV files) and Wikidata. The resulting graph is then transformed in the data format required by the target application. However, the wide range of reusable components configurable within a Chimera pipeline enables data retrieval from heterogeneous information systems, including systems based on virtualisation or federation.

3. Scenario

The proposed scenario exemplifies the challenges of implementing mobility-related services from heterogeneous data sources and how knowledge graphs can help address them [13]. The demo aims to develop a solution for collecting data to power an interactive dashboard that encourages public transport use by showcasing landmarks and points of interest near public transport stops. The solution should support multiple stakeholders in the mobility domain to contribute data independently while preserving interoperability and avoiding conflicts between datasets. As an example stakeholder, we consider public transport operators, which typically publish network and timetable information using the General Transit Feed Specification (GTFS)⁶. GTFS is an open standard for describing transit systems through structured datasets containing information such as stops, routes, trips, and schedules. To enrich the transit data, information about nearby landmarks should be retrieved from Wikidata through queries executed against the Wikidata Query Service. Finally, the enriched data should be forwarded to an external service that populates a dashboard where public transport stops and landmarks are represented by markers of different colours. In the demo, every user can test the solution by selecting a publicly available GTFS feed⁷. The implemented integration processes the GTFS feed and adds new data to the dashboard by combining transit and landmark information, creating a richer representation of the urban environment. The technical implementation is performed using the Chimera framework and is described in Section 4.

4. Implementation

The described scenario is implemented in a demo as a set of Chimera pipelines, shown in Figure 1 and available on GitHub⁸. The input GTFS feed is first transformed into an RDF knowledge graph, which is subsequently enriched with data retrieved from Wikidata. Finally, the resulting graph is then transformed into the application-specific format required by the dashboard.

4.1. RDF Knowledge Graph Construction

The input consists of a GTFS Schedule feed distributed as a ZIP archive. The Camel *File* component is used to read the feed and unzip the files. Relevant information about public transport stops is extracted and lifted into RDF through an MTL mapping executed by Chimera's *Mapping Template* component. The mapping transforms the tabular GTFS representation into a knowledge graph based on a reference semantic model, where stops are represented as geographic features together with their labels and coordinates. Geographic positions are lifted to a GeoSPARQL representation using `geo:hasGeometry`

⁵[https://github.com/cefriel/mapping-template/wiki/Mapping-Template-Language-\(MTL\)](https://github.com/cefriel/mapping-template/wiki/Mapping-Template-Language-(MTL))

⁶<https://gtfs.org/>

⁷A large open-source catalogue is maintained by MobilityData at <https://mobilitydatabase.org/>

⁸<https://github.com/cefriel/kg4di/tree/v1.0/camel-routes-exercises/e2>

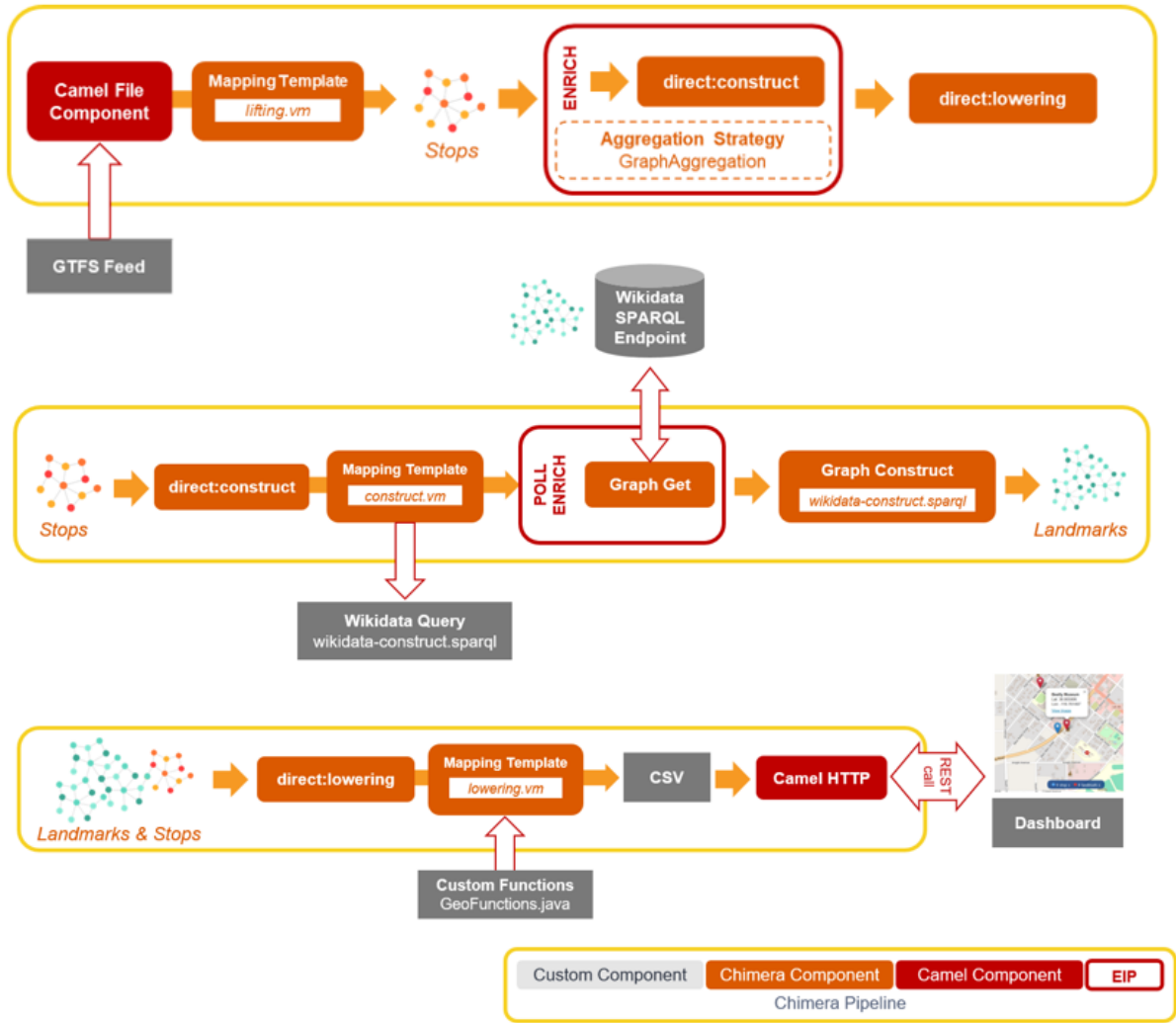


Figure 1: Diagram representing the Chimera pipelines implemented for the considered scenario

and `geo:asWKT`, which produces the WKT literals required by the Wikidata `wikibase:around` service. These geometries, as described in subsection 4.2 are then injected into dynamically generated SPARQL queries used to retrieve nearby landmarks and points of interest from Wikidata. The RDF graph produced in this phase serves not only as a harmonised representation of the source data but also as the basis for the enrichment process.

4.2. Knowledge Graph Enrichment and Aggregation

For the query to be used on Wikidata for the enrichment process, rather than executing a predefined SPARQL query, we define a Chimera pipeline, shown in Figure 2 that uses a second MTL mapping to dynamically generate a SPARQL `CONSTRUCT` query from the lifted graph, shown on the left side of Figure 3. The generated query, shown on the right side of Figure 3, embeds the coordinates of the transport stops obtained from the GTFS data and exploits Wikidata’s geospatial services to identify nearby landmarks and points of interest. This demonstrates how MTL can be used not only for conventional data transformation, targeting common formats such as JSON or CSV but also for the generation of SPARQL queries.

The generated `CONSTRUCT` query is executed against the Wikidata Query Service⁹ through Chimera’s Graph component, producing a second RDF graph containing nearby geographical entities together

⁹<https://query.wikidata.org/>

with selected information, such as labels, coordinates, and image references. The first graph and the Wikidata graph are then combined using the enrich EIP and Chimera's graph aggregation capabilities to produce a single integrated knowledge graph.

The aggregated RDF graph can be potentially stored as an intermediate artefact to support the construction of a larger KG, illustrating a key advantage of using RDF as the common representation for the integration pipeline. Additionally, this approach decouples data acquisition from downstream analysis and visualisation, while also facilitating extensibility. Other data sources can be integrated by configuring the corresponding endpoint and adapting the pipeline, without affecting either the GTFS ingestion workflow or the downstream components.

```
- beans:
  - name: constructTemplate
    type: com.cefriel.util.ChimeraResourceBean
    properties:
      url: "file:///mappings/construct.vm"
      serializationFormat: "vttl"

- route:
  from:
    uri: "direct:construct"
  steps:
    - to:
      uri: "mapt://rdf"
      parameters:
        template: '#bean:constructTemplate'
        basePath: './'
        fileName: 'wikidata-construct.sparql'
    - to:
      uri: "direct:wikidata"
```

Figure 2: Chimera YAML route used to produce the SPARQL construct query

4.3. From the Knowledge Graph to the Dashboard

In the final stage, the integrated knowledge graph is lowered into the CSV format required by the dashboard through an MTL mapping executed by the *Mapping Template* component. The mapping extracts information from the knowledge graph through a SPARQL query, obtaining both public transport stops and nearby landmarks, generating records containing their labels, coordinates, type, and, when available, image links. The produced dataset is delivered to the endpoint exposed by the dashboard application via the Apache Camel *HTTP* component and proper authentication configuration. The dashboard visualises the integrated information on an interactive map, where transport stops and Wikidata landmarks are represented through different markers and enriched with contextual information retrieved during the integration process.

4.4. Deployment

The demo is packaged to be reproducible with minimal setup. The Chimera runtime, route definitions, templates, custom functions, and dashboard can be launched in any containerised environment. Users can add GTFS ZIP archive in the configured input folder, execute the pipelines, and inspect the resulting map through a locally hosted web interface. Detailed instructions and alternative execution approaches are available in the README 8.

```

#set ($positions = $reader.getDataframe("
SELECT ?position ?featureCoord
WHERE {
  ?position a tmjou:ScheduledStopPoint ;
           geo:hasGeometry [ geo:asWKT ?featureCoord ] .
}
ORDER BY ?position
LIMIT 10
"))

CONSTRUCT {
  ?place a wd:Q618123 ;
        rdfs:label ?placeLabel ;
        wdt:P625 ?coord ;
        schema:image ?image ;
        geo:hasGeometry [ geo:asWKT ?coord ] .
}
WHERE {

  #foreach($position in $positions)
  {
    SERVICE wikibase:around {
      ?place wdt:P625 ?coord .
      bd:serviceParam wikibase:center
"$position.featureCoord"^^geo:wktLiteral .
      bd:serviceParam wikibase:radius "0.5" .
    }
  }
  #if($foreach.hasNext) UNION #end
#end

SERVICE wikibase:label {
  bd:serviceParam wikibase:language "en".
}
}
LIMIT 25

```

```

CONSTRUCT {
  ?place a wd:Q618123 ;
        rdfs:label ?placeLabel ;
        wdt:P625 ?coord ;
        schema:image ?image ;
        geo:hasGeometry [ geo:asWKT ?coord ] .
}
WHERE {

  {
    SERVICE wikibase:around {
      ?place wdt:P625 ?coord .
      bd:serviceParam wikibase:center
"Point(-116.40094 36.641496)"^^geo:wktLiteral .
      bd:serviceParam wikibase:radius "0.5" .
    }
  }
  UNION
  {
    SERVICE wikibase:around {
      ?place wdt:P625 ?coord .
      bd:serviceParam wikibase:center
"Point(-116.784582 36.868446)"^^geo:wktLiteral .
      bd:serviceParam wikibase:radius "0.5" .
    }
  }
  UNION
  {
    SERVICE wikibase:around {
      ?place wdt:P625 ?coord .
      bd:serviceParam wikibase:center
"Point(-116.81797 36.88108)"^^geo:wktLiteral .
      bd:serviceParam wikibase:radius "0.5" .
    }
  }
  UNION
  [...]
}

```

Figure 3: On the left, MTL mapping used to produce the SPARQL Construct query executed on Wikidata. On the right, resulting SPARQL query obtained for specific stop places.

5. Potential Impact for Building Wikidata Applications

The proposed approach supports the development of Wikidata-based applications by separating enrichment logic from application code. Rather than querying Wikidata directly from the user-facing application, developers can define a reusable semantic integration pipeline whose output is consumed by the target system. In the demonstrated use case, the dashboard does not need to handle GTFS parsing, Wikidata query generation, or RDF graph aggregation; it only consumes the final CSV representation produced by the pipeline. This reduces coupling and makes the enrichment process explicit, configurable, and easier to maintain.

The use of RDF as an intermediate representation also enables a more transparent and data-driven integration process. Local data that are not natively represented as RDF, such as CSV files, JSON documents, API responses, or domain-specific archives, can first be lifted into a knowledge graph and then used to guide Wikidata enrichment. In the demo, for example, the coordinates extracted from the GTFS feed determine the spatial search performed on Wikidata. The resulting intermediate graph can be inspected, queried, serialised, validated, or further enriched before being lowered into the format required by the final application. This is useful when developers need to assess which Wikidata entities were retrieved, how they relate to local data, and whether additional filtering or ranking is needed.

Finally, the approach lowers the effort required to adapt Wikidata-based enrichment to different application scenarios. Since Chimera pipelines are assembled from declarative route definitions and templates, individual steps can be modified without rewriting the whole application. The same pattern can be adapted to another city by changing the input GTFS feed, to another category of Wikidata entities by modifying the CONSTRUCT template, or to another dashboard by replacing the lowering template. Although the demo focuses on mobility data, the same architecture can support applications in domains such as cultural heritage, tourism, logistics, environmental monitoring, and smart-city

services. Overall, the main impact of the proposed approach is to make Wikidata-based enrichment more modular, reproducible, and adaptable across domains.

6. Conclusions and Future Work

This demo paper presented the use of Chimera for building composable semantic transformation pipelines that integrate application-specific data with Wikidata.

The demo illustrates how Wikidata can be used not only as a standalone knowledge base to be queried by applications, but also as an enrichment source within a broader semantic interoperability workflow. By preserving RDF as an intermediate, queryable representation, the pipeline decouples input formats, external knowledge retrieval, and output generation. Moreover, the presented approach demonstrates that Wikidata-based enrichment can be implemented as a reusable pipeline rather than as ad hoc application code. The implementation combines standard Camel integration mechanisms with Chimera's RDF-aware components, allowing the same architecture to be adapted to other input formats, reference ontologies, enrichment endpoints, and target applications.

Further experiments will investigate how the same approach can be reused in other application domains where Wikidata can provide contextual knowledge for heterogeneous local datasets. Future work will also focus on automating pipeline configuration and mapping rules using AI agents.

Acknowledgments

This work has been partially funded by the European Commission through the Chips Act Joint Undertaking project SMARTY (Grant no. 101140087).

Declaration on Generative AI

During the preparation of this work, the author(s) used Grammarly to improve grammar, check spelling, and reword. After using these tool(s)/service(s), the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

- [1] D. Vrandečić, M. Krötzsch, Wikidata: a free collaborative knowledgebase, *Commun. ACM* 57 (2014) 78–85. URL: <https://doi.org/10.1145/2629489>. doi:10.1145/2629489.
- [2] M. Grassi, M. Scrocca, A. Carenini, M. Comerio, I. Celino, Composable Semantic Data Transformation Pipelines with Chimera, in: D. Chaves-Fraga, A. Dimou, A. Iglesias-Molina, U. Serles, D. V. Assche (Eds.), *Proceedings of the 4th International Workshop on Knowledge Graph Construction co-located with 20th Extended Semantic Web Conference*, volume 3471 of *CEUR Workshop Proceedings*, CEUR, Hersonissos, Greece, 2023. URL: <https://ceur-ws.org/Vol-3471/paper9.pdf>, ISSN: 1613-0073.
- [3] M. Scrocca, M. Grassi, A. Carenini, I. Celino, Knowledge Graphs for Data Interoperability with Chimera (KG4DI), 2026. URL: <https://zenodo.org/records/20542486>. doi:10.5281/zenodo.20542486.
- [4] Wikidata, Wikidata: SPARQL Query Service, https://www.wikidata.org/wiki/Wikidata:SPARQL_query_service, 2026. Accessed: 2026-07-14.
- [5] G. Lima, J. M. B. Rodrigues, M. Machado, E. Soares, S. R. Fiorini, R. Thiago, L. G. Azevedo, V. T. da Silva, R. Cerqueira, KIF: A Wikidata-Based Framework for Integrating Heterogeneous Knowledge Sources, *arXiv preprint arXiv:2403.10304* (2024). URL: <https://arxiv.org/abs/2403.10304>. doi:10.48550/arXiv.2403.10304.

- [6] D. Van Assche, T. Delva, G. Haesendonck, P. Heyvaert, B. De Meester, A. Dimou, Declarative RDF graph generation from heterogeneous (semi-)structured data: A systematic literature review, *Web Semant.* 75 (2023). doi:10.1016/j.websem.2022.100753.
- [7] J. Arenas-Guerrero, M. Scrocca, A. Iglesias-Molina, J. Toledo, L. Pozo-Gilo, D. Doña, Ó. Corcho, D. Chaves-Fraga, Knowledge graph construction with R2RML and RML: an ETL system-based overview, in: D. Chaves-Fraga, A. Dimou, P. Heyvaert, F. Priyatna, J. F. Sequeda (Eds.), *Proceedings of the 2nd International Workshop on Knowledge Graph Construction co-located with 18th Extended Semantic Web Conference (ESWC 2021)*, Online, June 6, 2021, volume 2873 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2021. URL: <https://ceur-ws.org/Vol-2873/paper11.pdf>.
- [8] M. Scrocca, M. Comerio, A. Carenini, I. Celino, Turning Transport Data to Comply with EU Standards While Enabling a Multimodal Transport Knowledge Graph, in: *The Semantic Web – ISWC 2020*, Springer, 2020. doi:10.1007/978-3-030-62466-8_26.
- [9] M. Scrocca, A. Carenini, M. Grassi, M. Comerio, I. Celino, Not everybody speaks RDF: Knowledge conversion between different data representations, in: *Fifth International Workshop on Knowledge Graph Construction@ ESWC2024*, 2024. URL: <https://ceur-ws.org/Vol-3718/paper3.pdf>.
- [10] Z. Gu, F. Corcoglioniti, D. Lanti, A. Mosca, G. Xiao, J. Xiong, D. Calvanese, A systematic overview of data federation systems, *Semantic Web* 15 (2024) 107–165. doi:10.3233/SW-223201.
- [11] G. Hohpe, B. Woolf, *Enterprise integration patterns: Designing, building, and deploying messaging solutions*, Addison-Wesley Professional, 2004.
- [12] A. Iglesias-Molina, D. Van Assche, J. Arenas-Guerrero, B. De Meester, C. Debruyne, S. Jozashoori, P. Maria, F. Michel, D. Chaves-Fraga, A. Dimou, The RML Ontology: A Community-Driven Modular Redesign After a Decade of Experience in Mapping Heterogeneous Data to RDF, in: T. R. Payne, V. Presutti, G. Qi, M. Poveda-Villalón, G. Stoilos, L. Hollink, Z. Kaoudi, G. Cheng, J. Li (Eds.), *The Semantic Web – ISWC 2023, Lecture Notes in Computer Science*, Springer Nature Switzerland, Cham, 2023, pp. 152–175. doi:10.1007/978-3-031-47243-5_9.
- [13] M. Scrocca, M. Grassi, M. Comerio, V. A. Carriero, T. Delgado Dias, A. Vieira Da Silva, I. Celino, Intelligent urban traffic management via semantic interoperability across multiple heterogeneous mobility data sources, 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. 218–235. doi:10.1007/978-3-031-77847-6_12.