

BioHackathon series:  
[BioHackathon Japan 2025](#)  
Mie, Japan, 2025  
[PURL Resolver](#)

Submitted: 19 Sep 2025

**License:**  
Authors retain copyright and  
release the work under a Creative  
Commons Attribution 4.0  
International License ([CC-BY](#)).

Published by [BioHackrXiv.org](#)

# A Lightweight PURL Resolver for Linked Life Science Data

Julia Koblitz <sup>1</sup> and Shuichi Kawashima <sup>2</sup>

<sup>1</sup> Leibniz Institute DSMZ-German Collection of Microorganisms and Cell Cultures <sup>2</sup> Database Center for Life Science, Joint Support-Center for Data Science Research, Research Organization of Information and Systems

## Abstract

Knowledge graphs in the life sciences are increasingly published using the Resource Description Framework (RDF) and queried via SPARQL endpoints. While these technologies enable powerful data integration, the identifiers returned in SPARQL results often do not resolve to meaningful resources, leaving users with non-actionable links. To address this issue, we developed a lightweight Persistent Uniform Resource Locator (PURL) resolver during the BioHackathon Japan 2025. The resolver is implemented in PHP, chosen for its ubiquity on standard web servers and its compatibility with the EasyRDF library for RDF handling. It is easy to configure, requires minimal maintenance, and supports both database redirects and ontology term rendering with content negotiation for RDF serializations. The system is available as open-source software (<https://github.com/JKoblitz/purl-resolver>) and deployed at <https://purl.dsmz.de>, where it now resolves most identifiers from the DSMZ Digital Diversity SPARQL endpoint (<https://sparql.dsmz.de>). Database IRIs lead to the corresponding web interfaces, ontology IRIs from the DSMZ Digital Diversity Ontology render directly as term pages, and unmapped entities are delegated to database-side resolvers. This approach enhances the usability of knowledge graphs by ensuring that all identifiers remain actionable for both humans and machines.

## Introduction

The Resource Description Framework (RDF) is a World Wide Web Consortium (W3C) standard designed to represent information in a graph-based structure of subject–predicate–object triples (Lassila & Swick, 1999). RDF provides the foundation for interoperable data integration across domains, and its power unfolds through SPARQL, the query language standardized by the W3C for retrieving and manipulating RDF data (Prud'hommeaux & Seaborne, 2008). Persistent Uniform Resource Locators (PURLs) are a complementary technology to ensure that identifiers expressed as Internationalized Resource Identifiers (IRIs) remain resolvable and stable over time (Shafer, 1995). Together, these technologies enable the construction of sustainable and FAIR (Findable, Accessible, Interoperable, Reusable) knowledge graphs that allow researchers to query, combine, and interpret data from heterogeneous sources.

Due to the rise of AI technologies, an increasing number of life science databases provide RDF and SPARQL endpoints to expose their data to the community. However, when users explore these graphs interactively, for instance by clicking on IRIs returned in a SPARQL query, they often encounter dead ends: the IRIs identify resources but do not resolve to human-readable or machine-readable representations. And in instances where they do, the results are often hard to interpret and sometimes links are not properly maintained. This disconnect between linked data identifiers and their usability in practice hampers exploration and reduces the perceived value of the linked data infrastructure.

## Motivation

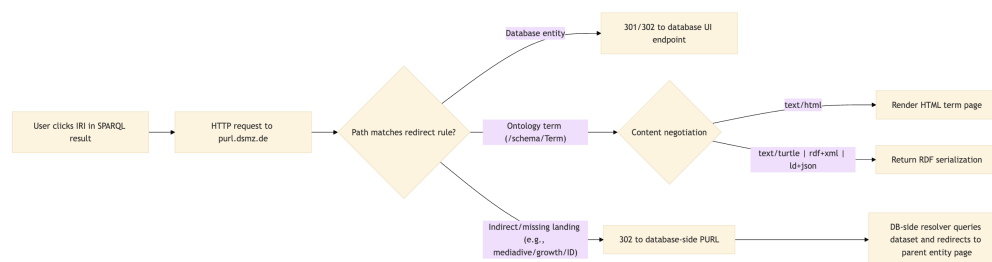
In the [DSMZ Digital Diversity](#) project, we maintain multiple biological databases such as *BacDive*, *MediaDive*, and *BRENDA*, all of which are available through RDF and SPARQL endpoints (Chang et al., 2020; Koblitz et al., 2022; Schober et al., 2024). While the linked data model provides globally unique identifiers, many of these IRLs do not resolve to meaningful resources for end users. For example, clicking on an IRI representing a strain or a culture medium in a SPARQL result would ideally lead the user to the corresponding page in the relevant database, or to a documentation page of an ontology term. Without this functionality, the user experience is interrupted, and the adoption of SPARQL endpoints as practical tools is reduced.

During the BioHackathon Japan 2025, our group set out to close this gap by developing a PURL resolver that ensures actionable IRLs across the DSMZ Digital Diversity knowledge graph. The goal was to design a solution that is simple to deploy, lightweight to maintain, and usable not only by our group but also by other database providers facing the same challenge.

## Idea

We decided to implement the resolver in PHP. Our rationale was pragmatic: PHP is supported by default in most standard web server environments, whether based on Apache or Nginx. This reduces the operational burden compared to frameworks requiring containerization or additional runtime environments. Furthermore, the EasyRDF library provides a convenient and well-tested abstraction layer for parsing, handling, and serializing RDF data (Miles et al., 2009).

The resolver acts as a central routing layer for all DSMZ Digital Diversity PURLs. Requests arriving at <https://purl.dsmz.de> are evaluated against a set of configurable patterns. If the request corresponds to a database entity, the resolver issues a redirect to the appropriate database interface. If the request corresponds to an ontology term from the DSMZ Digital Diversity Ontology (D3O), the resolver either serves an HTML documentation page for human users or a machine-readable RDF serialization, depending on the client's *Accept* header. Finally, if a PURL does not have a direct landing page, such as a growth dataset in *MediaDive*, the resolver redirects to a PURL endpoint implemented within the target database. There, a lightweight script performs the required mapping, for example linking a growth dataset back to its parent medium, before presenting the final resource page to the user.



**Figure 1:** Workflow diagram of the PURL resolver. Depending on the type of IRI requested, the resolver either redirects to the corresponding database entry, renders an ontology term, or forwards to a database-side resolver for indirect entities.

## Results

The PURL resolver was implemented, documented, and published on GitHub, where it is available for community use under an open license:

<https://github.com/JKoblitz/purl-resolver>

Installation requires only the standard PHP ecosystem and dependency management through Composer, making it straightforward for other groups to adopt. To demonstrate its practical utility, we deployed the resolver at <https://purl.dsmz.de>. Since deployment, most PURLs from the DSMZ Digital Diversity SPARQL endpoint (<https://sparql.dsmz.de>) are now actionable.

The resolver delivers several concrete benefits. Database IRIs now resolve seamlessly: for example, <https://purl.dsmz.de/bacdiv/strain/123> redirects to <https://bacdiv.dsmz.de/strain/123>. Ontology IRIs now serve human-readable term pages, such as <https://purl.dsmz.de/schema/DNASequence>, featuring sub and superclasses and descriptions, while also supporting content negotiation for RDF formats like Turtle, RDF/XML, or JSON-LD. For complex entities without direct landing pages, such as growth records in *MediaDive*, the resolver redirects to a PURL endpoint within *MediaDive* itself, where further logic resolves the dataset to the most appropriate medium page. This hybrid approach ensures that all identifiers remain meaningful entry points for both humans and machines.



## DNASequence

<https://purl.dsmz.de/schema/DNASequence>

Type: owl:Class

Subclass of: [Nucleotide Sequence](#)

A sequence of nucleotides within DNA that encodes genetic information. DNA sequences are responsible for heredity and govern cellular functions through gene expression.

Subclasses:

- [Gene Sequence](#)
- [Genome Sequence](#)

Machine-readable: [Turtle](#) • [JSON-LD](#) • [RDF/XML](#)

**Figure 2:** Screenshot of a term page generated by the PURL resolver. The example shows the ontology class “Strain” from the DSMZ Digital Diversity Ontology (D3O), with its label, definition, and relationships rendered as human-readable HTML.

## Discussion

Persistent identifiers are a cornerstone of sustainable linked data infrastructures. By combining simplicity of deployment with robust functionality, the resolver developed here provides a practical model that can be reused by other data providers. Unlike large-scale resolver infrastructures such as [identifiers.org](https://identifiers.org) (Juty et al., 2011), our approach focuses on lightweight maintainability in small to medium-scale deployments, which is often the reality for specialized biological databases.

The use of PHP may seem unusual in contemporary web development, but it provides distinct advantages in this context. PHP is widely supported, has a low barrier to entry, and ensures

that the resolver can be hosted on standard institutional web servers without the need for complex deployment pipelines. This lowers adoption thresholds and empowers smaller groups to offer resolvable PURLs with minimal infrastructure overhead.

## Acknowledgements

We gratefully acknowledge the organisers of the [DBCLS BioHackathon 2025](#) for providing an inspiring and collaborative environment in Mie, Japan. The intensive hackathon setting enabled rapid prototyping and direct feedback from peers, which proved essential to the success of this project.

Chang, A., Jeske, L., Ulbrich, S., Hofmann, J., Koblitz, J., Schomburg, I., Neumann-Schaal, M., Jahn, D., & Schomburg, D. (2020). BRENDA, the ELIXIR core data resource in 2021: New developments and updates. *Nucleic Acids Research*, 49(D1), D498–D508. <https://doi.org/10.1093/nar/gkaa1025> [cito:citesAsDataSource]

Juty, N., Le Novère, N., & Laibe, C. (2011). Identifiers.org and MIRIAM registry: Community resources to provide persistent identification. *Nucleic Acids Research*, 40(D1), D580–D586. <https://doi.org/10.1093/nar/gkr1097> [cito:citesAsRelated]

Koblitz, J., Halama, P., Spring, S., Thiel, V., Baschien, C., Hahnke, R. L., Pester, M., Overmann, J., & Reimer, L. C. (2022). MediaDive: The expert-curated cultivation media database. *Nucleic Acids Research*, 51(D1), D1531–D1538. <https://doi.org/10.1093/nar/gkac803> [cito:citesAsDataSource]

Lassila, O., & Swick, R. R. (1999). *Resource Description Framework (RDF) Model and Syntax Specification*. <https://www.w3.org/TR/1999/REC-rdf-syntax-19990222/> [cito:citesAsAuthority]

Miles, M., Bechhofer, S., Prud'hommeaux, E., Seaborne, A., Carroll, J. J., Dickinson, I., Gibbins, N., Klyne, G., McBride, B., Newman, C. others. (2009). *EasyRDF: A PHP library for working with RDF*. <https://www.easyrdf.org/> [cito:usesMethodIn]

Prud'hommeaux, E., & Seaborne, A. (2008). *SPARQL Query Language for RDF*. <https://www.w3.org/TR/rdf-sparql-query/> [cito:citesAsAuthority]

Schober, I., Koblitz, J., Sardà Carbasse, J., Ebeling, C., Schmidt, M. L., Podstawka, A., Gupta, R., Ilangoan, V., Chamanara, J., Overmann, J., & Reimer, L. C. (2024). BacDive in 2025: The core database for prokaryotic strain data. *Nucleic Acids Research*, 53(D1), D748–D756. <https://doi.org/10.1093/nar/gkac959> [cito:citesAsDataSource]

Shafer, K. (1995). *Introduction to Persistent Uniform Resource Locators*. [https://web.archive.org/web/20160103053338/http://www.isoc.org/inet96/proceedings/a4/a4\\_1.htm](https://web.archive.org/web/20160103053338/http://www.isoc.org/inet96/proceedings/a4/a4_1.htm) [cito:citesAsAuthority]