Bravo MaRDI: A Wikibase Knowledge Graph on Mathematics

Moritz Schubotz¹, Eloi Ferrer², Johannes Stegmüller¹, Daniel Mietchen¹, Olaf Teschke¹, Larissa Pusch² and Tim Conrad²

¹FIZ Karlsruhe – Leibniz Institute for Information Infrastructure, Berlin, Germany
²Zuse Institute Berlin, Berlin, Germany

Abstract
Mathematical world knowledge is a fundamental component of Wikidata. However, to date, no expertly curated knowledge graph has focused specifically on contemporary mathematics. Addressing this gap, the Mathematical Research Data Initiative (MaRDI) has developed a comprehensive knowledge graph that links multimodal research data in mathematics. This encompasses traditional research data items like datasets, software, and publications and includes semantically advanced objects such as mathematical formulas and hypotheses. This paper details the abilities of the MaRDI knowledge graph, which is based on Wikibase, leading up to its inaugural public release, codenamed Bravo, available on https://portal.mardi4nfdi.de.

This paper is submitted to the ISWC Wikidata Workshop 2023 novel resource track.

1. The One-Stop-Portal Vision

The Mathematical Research Data Initiative (MaRDI) [10] strives to bring the FAIR data principles [23] to life for mathematics and its research data (MathRD). A critical aspect of this ambition involves creating a unified portal¹ that offers comprehensive access to all open research data within mathematics and related fields.

In this manuscript, we introduce the first Bravo release of the portal, now freely accessible through direct access or various APIs. Rather than constructing such a portal from scratch - a tactic adopted by parallel initiatives in Germany’s National Research Data Initiative (NFDI) - we chose to utilize Wikimedia technology, predominantly MediaWiki, as a foundation, for several reasons:

• MediaWiki offers a thriving open-source environment with a myriad of free and open-source software.

• A considerable user base exists for MediaWiki; many mathematicians are already acquainted with Wikipedia and, to some extent, Wikidata.

¹Analogous initiatives name this type of service as ‘Single Point of Access’ [11].
• Upon the project’s completion, we have the option to incorporate our data into Wikidata or another compatible platform.

• Both the Wikimedia mathematics community and portal users, notably mathematicians based in Germany, mutually benefit from our advancements.

• We seamlessly inherit knowledge from Wikipedia and Wikidata, painstakingly curated by hundreds of thousands of contributors over the past two decades.

However, this decision comes with challenges; becoming part of the Wikimedia movement necessitates embracing their work culture and guidelines and engaging with the community to achieve consensus on sustainable solutions. Our contributions in this context are as follows:

1. We deliver a current ecosystem to house Wikibase and associated services on-premises.

2. We customize the general-purpose Wikibase infrastructure to cater to the specific needs of the mathematical community.

3. We have devised a mechanism for importing external data that aligns with the Wikidata data model.

4. We have actively enhanced the MediaWiki ecosystem by expanding and improving services such as QuickStatements, MediaWiki extensions like Math and MathSearch, and libraries like the Wikibase Integrator.

5. We have seeded the MaRDI KG with nearly 100 million triples\(^2\) aggregated from various sources.

This paper is organized as follows: Section 2 begins with an overview of our technology stack. This is followed by Section 3, which presents our mathematics-specific extensions. Subsequently, Section 4 details the infrastructure for seeding and updating the knowledge graph. In Section 5, we illustrate the functionality of our system with an example. Finally, Section 6 provides a conclusion and discusses future prospects.

### 2. Technical Infrastructure Overview

The MaRDI portal’s infrastructure was initially constructed using the Wikibase release pipeline. The release pipeline incorporates a docker-compose file that delineates several services, including MediaWiki with Wikibase extensions, the Wikidata Query Service, and the Blazegraph Knowledge Graph backend. The docker-compose file allows for the launching of a pre-set group of services via a single command, thus optimizing reproducibility and system or web host independence.

The MaRDI portal operates on a solitary open-stack instance with 32 GB of main memory and 8 CPUs, alongside a 300 GB HDD. Only Docker and Git are installed on the host system. All

\(^2\)Refer to the latest statistics from [https://portal.mardi4nfdi.de/wiki/Portal](https://portal.mardi4nfdi.de/wiki/Portal).
configurations are housed in partially encrypted Git repositories, which feature a CI pipeline enabling direct deployment to production without necessitating access to the actual machine. If the data volume increases, we have the capability to transition from a single-node deployment to a Docker swarm with minimal modifications to our service specifications.

The following provides a concise description of selected services from our infrastructure, which currently comprises 28 services [8]:

**Traefik** As the primary entry point to our portal, all external network traffic is channeled through the Traefik reverse proxy. Traefik is responsible for managing SSL certificates properly via in-built lets-encrypt management and protective measures to guard individual services against unauthorized use. Additionally, Traefik collates access logs, enabling centralized usage examinations for our interconnected services.

**MediaWiki** In order to maintain compatibility with Wikipedia and manage only one version of mathematics-specific extensions, we chose to fork the infrastructure and update the container with the same versions utilized by Wikipedia in production starting from a previous effort [4].

**WDQS** WDQS is the abbreviation of Wikidata Query Service which we have adapted to query the MaRDI portal and renamed it to MaRDI Query Service. This service is deployed based on four Docker containers: the Blazegraph database storing the knowledge graph, an updater that synchronizes the Wikibase entries with the Blazegraph backend, a frontend GUI to perform SPARQL queries, and a proxy to protect the SPARQL endpoint. The GUI for this service can be accessed at query.portal.mardi4nfdi.de.

**LaTeXML** LaTeXML is a service that converts LaTeX input to content and presentation MathML. It is employed for MathSearch.

**BaseX** BaseX is an XML database utilized for formula search. The MathSearch extension is responsible for keeping this service current.

**Backup** A straightforward, custom backup process was installed to preserve precisely the data we wish to retain, including the information saved on the SQL database, the MediaWiki pages as XML and the uploaded files.

**Scholia** Scholia is a tool that facilitates scholarly exploration of subsets of Wikidata’s knowledge graph via a web browser [19]. Its backend queries the Wikidata Query Service through predefined SPARQL queries parametrized by Wikidata identifiers, and its frontend displays the results of several of these queries as profiles. A multitude of such profile types exist, including works, venues, topics, authors, awards, events, or organizations. MaRDI’s instance is deployed via a fork [3] of the original [2] deployed on the Wikimedia Toolforge. The intent behind the fork is to ensure maximum customizability for mathematics, with general improvements fed back upstream wherever possible. The

---

3[https://scholia.portal.mardi4nfdi.de/](https://scholia.portal.mardi4nfdi.de/)
4[https://scholia.toolforge.org/](https://scholia.toolforge.org/)
ongoing customization includes efforts towards federated queries that utilize both the MaRDI knowledge graph and Wikidata.

**Portainer** Portainer is a management tool for Docker containers, images, volumes, and networks that operates via a web interface. It provides multiple ways of interacting with containers, including viewing their logs and accessing the container console. Portainer is compatible with standalone Docker installations, Docker Swarm, and Kubernetes.

**Watchtower** Watchtower is a tool that keeps Docker images up to date and guarantees the early installation of critical patches, provided Docker containers follow semantic versioning.

**Prometheus & Grafana** Prometheus is a monitor and alerting toolkit to monitor the health of the portal and the infrastructure. We visualize the data gathered by Prometheus using the dashboarding tool Grafana.

To guard against unintended side effects arising from changes to this multifaceted service infrastructure, we implemented a Testing concept that is triggered by any modification. This helps increase our confidence in not deploying erroneous code to our production environment. Additionally, Identity and Access Management (NFDIIAM) is a crucial aspect. MaRDI utilizes OAuth as a protocol, which is also supported by the NFDI’s wide basic service, NFDIIAM.

### 3. Mathematics-specific Extensions

Our portal leverages two math-specific MediaWiki extensions, namely, Extension-Math [26, 22, 15, 14] and Extension-MathSearch [28, 20, 17]. The former extension, deployed across all Wikimedia projects, including Wikipedia, facilitates the rendering of mathematical expressions. The latter makes mathematical expressions searchable and links knowledge graph data to HTML5.

#### 3.1. Rendering of Mathematical Expressions

The MaRDI portal, alongside Wikipedia and Wikidata, employs texvc markup to express mathematical formulas. This LaTeX variant does not permit runtime syntax changes, favoring context-free formula processing with a regular grammar. Mathematical expressions in wikitext are denoted by the wikitext tag `<math>` (to be distinguished from the HTML5 element `<math>`). These `<math>` tags enclose the texvc expressions, which are then rendered. In contrast, Wikibase introduces a data type specifically for mathematical expressions. Examples of such notations can be found in the DLMF defining formula in the MaRDI-KG [24] and in math-tagged elements in Wikitext.

Typically, portal users view page content in their web browsers. The Extension-Math processes the texvc math from the Wikttext or the Wikibase entities into HTML, allowing the

---

5https://portal.mardi4nfdi.de/wiki/Item:Q1799
6https://portal.mardi4nfdi.de/wiki/Non-negative_Matrix_Factorization_for_Time-Resolved_Raman_Spectroscopy_Data
browser to display the rendered formulas correctly. This rendering pipeline leverages an external
REST-based service to generate SVG images from texvc within the MediaWiki ecosystem [26].

As of 2023, all Chromium-based browsers support MathML, a web standard defined by the
W3C. MathML is part of HTML5, akin to SVG, and enables native math rendering without
needing images or browser extensions. Thus the latest versions of Edge, Chrome, Firefox,
and Safari fully support HTML5, including MathML since January 2023. Due to changes
in the MediaWiki ecosystem shifting away from microservices, and to augment processing
capabilities for semantic annotation, the MaRDI Project developed a rendering pipeline written
in pure PHP [5]. It generates MathML directly within the Math extension without the need for
communication with REST-based services.

Initially, TexVC leverages a parsing expression grammar to extract a language-independent
parse tree from LaTeX formulas. This parse tree verifies the formula’s syntactical correctness
and provides feedback to authors in edit mode [12]. To validate this component’s functionality,
we used 312,190 formulas extracted from the English Wikipedia [27].

The parse tree then undergoes root-first traversal, translating each item to a corresponding
MathML item. The native MathML component creates the correct MathML representation for
the 724 LaTeX commands supported by TexVC. The MathML used for automatic validation is
generated from Mathoid [26] and LaTeXML [29].

The MaRDI project contributed to the open-source community by developing a PHP version of
the TypeScript-based mhchemParser [1, 13], integrating it into the MediaWiki ecosystem
[9]. The mhchem syntax is employed alongside the mathematical formula notation to write
chemical equations within the Wikibase ecosystem. Implementing this component in PHP
enables the processing of these equations within the PHP-based math processing pipeline. The
component’s functionality was validated using 116 automated tests, which incorporate the

The native MathML pipeline in the MaRDI Wikibase paves the way for further advancements,
such as parsing formulas from arXiv, semantic annotation of formulas, enhancing accessibility
for visually impaired users, and resolving disambiguation scenarios.

3.2. Formula Search in the MaRDI Portal

The MaRDI Portal employs the MathSearch extension to search mathematical expressions
based on texvc input. The original version utilized the Math Web Search engine, also used in
the formula search of zbMATH Open [21]. In the MaRDI project, we optimized the formula
search components for the Docker-based infrastructure running the latest MediaWiki versions.
The MaRDI setup uses the BaseX database to store the search index created for all formulae
in the MaRDI portal.

---

7https://www.w3.org/TR/MathML/
8https://www.lambdatest.com/web-technologies/mathml
9https://phabricator.wikimedia.org/T303822
10https://gerrit.wikimedia.org/r/c/mediawiki/extensions/Math/+/923597
11https://texdoc.org/serve/mhchem/0
12https://www.mediawiki.org/wiki/Extension:MathSearch
13https://github.com/MaRDI4NFDI/formulasearch
4. Seeding Mathematical Research Data

4.1. Defining Mathematical Research Data (MathRD)

In the realm of mathematics, research data (MathRD) [16, 6] encompasses all data forms integral to the research process. These data forms facilitate the creation and analysis of mathematical models, substantiate proofs, test algorithms, and generally elucidate mathematical phenomena. MathRD presents itself in various types, structures, and degrees of accessibility and interoperability.

MathRD ranges from highly structured forms like symbolic or numerical data, used for encapsulating theorems, proofs, number sequences, or matrices. Geometric data, delineating objects like curves, surfaces, or polytopes, also fall into this category. In contrast, MathRD may also include less structured forms such as mathematical models or observational data. At the extreme end of the spectrum, plain text data like scientific papers, online resources, articles, and books are also encompassed under MathRD.

The MaRDI knowledge graph is engineered to effectively capture the intricate structure of this data. It does this by aggregating metadata from diverse sources, each describing mathematical research data, into a consolidated graph.

4.2. Constructing the MaRDI Knowledge Graph

As previously alluded to in Section 1, the data model for the MaRDI knowledge graph is an extension of the existing data model in Wikidata. Practically, this indicates that entities for the MaRDI knowledge graph are sourced from Wikidata if pre-existing there. Once integrated into the MaRDI knowledge graph, these entities can be further enriched with additional statements, though the detailed modalities of whether and how to keep the MaRDI data in sync with Wikidata remain to be worked out.

This approach is particularly vital in the case of properties, as it necessitates the design of our data model based on the available properties in Wikidata. New properties are defined only when a corresponding one is absent in Wikidata. As a result, many of the properties found in Wikidata are also present in the MaRDI knowledge graph. We anticipate the introduction of new properties as the mathematical knowledge graph expands, which will be specifically linked to mathematics.

To facilitate entity import from Wikidata, we have established an entity importer functionality built upon the Python module WikibaseIntegrator14. This module forms the basis for a Python class capable of transferring any given entity or list of entities from Wikidata to our Wikibase instance. This importer class copies not only the label, description, and aliases of the entity but also its statements, with the option to import additional connected entities. The importer class includes a parameter that determines the desired depth level of the import. By default, one level of depth is selected, importing all statements for each imported entity, and importing only label, description, and aliases for the entities mentioned in these statements.

The MaRDI knowledge graph further includes properties such as wikidata PID15 and wikidata

14https://github.com/LeMyst/WikibaseIntegrator
15https://portal.mardi4nfdi.de/wiki/Property:P11
QID\textsuperscript{16}, which store the Wikidata identifier for each imported property and item, respectively. This information is also stored in an internal SQL table, mapping the MaRDI knowledge graph identifiers to Wikidata identifiers. This table additionally incorporates a parameter for each entity indicating whether all its statements have been imported or only its label, description, and aliases.

This approach facilitates a comprehensive understanding of the overlap between the MaRDI knowledge graph and Wikidata at any given time. Moreover, it ensures synchronization between imported entities and Wikidata, and maintains consistency between the knowledge graphs, thus making the eventual integration of the MaRDI knowledge graph into Wikidata seamless.

The properties imported from Wikidata bolster the creation of new items in the MaRDI knowledge graph. In addition to the Wikidata items, metadata from eight distinct sources have been imported to varying degrees. These include formula metadata from DLMF, software metadata from swMATH, publication metadata from zbMATH Open, arXiv and crossref, metadata from resources stored at Zenodo, metadata on discrete geometric objects stored at polyDB\textsuperscript{17} and metadata on R packages published at CRAN\textsuperscript{18}.

4.3. Persistent Identifiers and Compliance with FAIR Principles

Adherence to FAIR principles necessitates the assignment of persistent identifiers to each linked resource in the MaRDI knowledge graph. It’s essential to differentiate between extrinsic and intrinsic identifiers:

- **Extrinsic identifiers**: These identifiers are linked to a specific resource through a register. DOIs, Wikibase IDs or ORCID IDs are instances of this category.

- **Intrinsic identifiers**: These identifiers are innately linked to the resource and do not necessitate an external register for association. An example is the name of a software package, if used as its identifier.

Every resource in the MaRDI knowledge graph is assigned an extrinsic identifier by being created as a Wikibase entity. These follow the format used in Wikidata, with item identifiers beginning with 'Q' and property identifiers with 'P'. Additionally, entities imported from a given source incorporate a statement that includes the identifier from the original source. Currently, the MaRDI knowledge graph supports the following extrinsic identifiers:

- DOI
- ORCID iD
- Digital Library of Mathematical Functions ID
- swMATH work ID

\textsuperscript{16}https://portal.mardi4nfdi.de/wiki/Property:P12
\textsuperscript{17}https://polydb.org/
\textsuperscript{18}https://cran.r-project.org/
• zbMATH Open document ID
• zbMATH author ID
• arXiv ID
• arXiv author ID
• Mathematics Subject Classification ID
• Zenodo ID

Further, two types of resources are linked to their source using extrinsic identifiers:

• CRAN Project: This identifier, imported from Wikidata\(^1\), links to an R package published at the Comprehensive R Archive Network. The identifier is a string that corresponds to the name of the R package.

• PolyDB ID: This identifier, created in the MaRDI knowledge graph, links to the original metadata source of a polyDB collection. The identifier string corresponds to the name of the collection.

5. Case study: Orthogonal Polynomials and Special Functions

In this section, we showcase some capabilities of the MaRDI portal using semantically enhanced Mathematical formulas as an example, specifically focusing on Orthogonal Polynomials and Special Functions (OPSF).

In the field of OPSF, relationships between functions are essential. The NIST Digital Library of Mathematical Functions, along with its companion, the Digital Repository of Mathematical Formulae \(^2\, 24\), is an indispensable source of identifiers and formulas pertaining to this field. Their main function is as aggregators of well-established knowledge, rather than providers of novel work.

A special variant of LaTeX has been developed by the team to enable a higher level of machine-readable semantics. For example, one typically writes \( i \) in LaTeX to represent the imaginary unit. In the context of DLMF/DRMF, however, one would use \( \text{iunit} \). Both versions render the same output. However, when the LaTeX source code is web-rendered rather than PDF-generated, the \( \text{iunit} \) version offers the added advantage of being clickable. This improves accessibility by linking to a human-readable explanation of what the imaginary unit represents.

In Figure 1a, \( \text{iunit} \) links to https://dlmf.nist.gov/1.9#E1, which is considered a permanent identifier. This means that the numbering remains consistent, irrespective of the introduction of additional material.

To make DLMF data accessible from the MaRDI portal, we have imported the formulas into our knowledge graph. Each formula has a corresponding Wikibase item. For example, the formula displayed in Figure 1a corresponds to Q3045 and links to the definition Q1399.

\(^{19}\)https://www.wikidata.org/wiki/Property:P5565
(a) Screenshot from the NIST Digital Library of Mathematical Functions (DLMF) https://dlmf.nist.gov/7.5 demonstrating the use of \(\iunit\).

(b) Representation of 1a in the MaRDI Wikibase Instance.

Figure 1: A DLMF formula in different representations.

Our SPARQL query interface enables users to retrieve all formulas that include a link to the imaginary unit with the following command\(^\text{20}\):

\[
\text{SELECT ?item WHERE { ?item wdt:P4 wd:Q1399 .}}
\]

More complex queries are also supported, such as searching for all formulas indirectly dependent on the gamma function.

We have additionally created formula homepages, similar to the DRMF project, to make the data available for classic searches and to offer a more convenient view compared to the standard Wikidata item view. Once the import of zbMATH articles is completed, bibliographic references related to the formula, software models, and algorithms will be linked from here.

\(^{20}\)https://tinyurl.com/2bd4jv8k
Our software can also translate the expression from DLMF to Maple and Mathematica [7]. However, due to potential licensing issues, we only display the formulas on the separate site https://lct.wmflabs.org, and not in the portal.

6. The road ahead

The Bravo Release of the MaRDI platform served as a proof of concept for leveraging MediaWiki and associated services from the Wikimedia ecosystem as a platform for the National Research Data Infrastructure, focusing on the field of mathematics. We successfully demonstrated the platform’s capability to handle domain-specific requirements, such as the manipulation of mathematical expressions, proving the adaptability of this general-purpose platform for specialized needs.

However, we recognize that several aspects of the platform’s usability require enhancement. The current interface design, reminiscent of a Wikipedia fork, needs to be reimagined to reflect the nuances of a research data management platform. Furthermore, while advanced users might find formulating SPARQL queries straightforward, the average user might need additional support. Although Wikidata provides visual aids to improve query generation, we aim to go a step further by integrating a conversational interface to facilitate SPARQL query generation following our previous MathQA efforts [18].

The platform’s citation capabilities also leave room for improvement. While MediaWiki has long provided permanent links (using the oldid flag), these could be further enhanced. For instance, we are contemplating the implementation of a feature that stores a snapshot of the PDF on Zenodo for convenience. However, the issue of citing query results remains an open research problem, posing a challenge for reproducibility in data-intensive mathematics.

The road ahead is not without its obstacles, but we are committed to continuous improvement and development. Our aim is to deliver a platform that not only adheres to the FAIR principles but also becomes an integral part of the mathematical research data ecosystem, thereby facilitating collaboration and advancement in mathematical research.

Acknowledgments

This work was funded by the MaRDI project [10] under DFG grant number 460135501. The text of this manuscript was improved with several AI tools, including Grammarly and ChatGPT.

References


