A Digital Preservation Wikibase

Katherine Thornton
Yale University Library
United States
katherine.thornton@yale.edu
0000-0002-4499-0451

Kenneth Seals-Nutt
Yale University Library
United States
kenneth.seals-nutt@yale.edu
0000-0002-5926-9245

Abstract – The Wikidata for Digital Preservation (WikiDP) Wikibase is a project of the Yale University Library's department of digital preservation. The WikiDP Wikibase is an open knowledge base that is publicly available on the web. We outline the relationship of Wikibase to other software of the Wikimedia Foundation, and provide examples of where it is being used. We describe the data models, data sources, and connections between the WikiDP Wikibase and the Wikidata knowledge base. We discuss our decision to use Wikibase for this project which involves transforming a data set related to software into a knowledge base using technologies of the Semantic Web.

Keywords – Wikibase, Wikidata, software metadata, Shape Expressions, Semantic Web,

Themes – Community, Exchange, Innovation

I. Introduction

We introduce Wikidata for Digital Preservation (WikiDP), a Wikibase instance related to the domain of computing. This knowledge base contains structured metadata about software, file formats, and configured software environments. Data can be searched via a search bar in the user interface, an application programming interface (API) and a SPARQL endpoint. The knowledge base is publicly available on the web [1].

The fact that this knowledge base is available on the web in a way that is accessible to both humans and machines enables collaboration between large numbers of people around this resource [1]. Making structured data available to machines is part of Tim Berners-Lee's vision for the Semantic Web [2]. Incorporating technologies of the Semantic Web in the field of digital preservation allows us to improve the interoperability of digital preservation systems with a broad landscape of other systems and data sources. This increases the utility and the value of our data [3]–[7].

Created in 2019, the WikiDP Wikibase contains data from the National Software Reference Library (NSRL) structured to support the description of configured software environments. It contains data about thousands of software titles including information about when they were published, who developed them, what operating systems they are compatible with, and the human languages in which they are available.

We designed the WikiDP Wikibase to support the work of the Emulation as a Service Infrastructure (EaaSI) program of work at Yale University Library [8]. The EaaSI program of work aims to provide a broad range of configured software environments using a range of software emulators. EaaSI users can then interact with legacy software titles which may require outdated operating systems, or other software, that may be inconvenient to access. The EaaSI team creates metadata descriptions for configured software environments and stores them in the WikiDP Wikibase.

We outline the steps we took to design and populate this Wikibase. We describe how we mapped the data in the WikiDP Wikibase to Wikidata, and share some example federated queries that allow us to ask questions of the WikiDP Wikibase and Wikidata at the same time.

II. Wikidata

Wikidata is a community-curated knowledge base of structured data [9]. Tens of thousands of volunteer editors contribute data to Wikidata relating to a broad range of topics [10]. Data published in Wikidata is available under a Creative Commons Zero (CC0) license. Anyone is free to reuse data from Wikidata for any purpose.

There are multiple options for data reuse from Wikidata. Data in Wikidata can be accessed via the API [2]. Data can also be accessed via SPARQL. SPARQL is a query language for RDF data [11]. RDF is an acronym for Resource Description Framework, a graph-based data model [12]. Wikidata has a SPARQL endpoint that allows anyone with access to the internet to submit queries and get results[3]. Users can select a format for downloading the results of a query. The available formats are JSON, TSV, CSV, HTML and SVG[4].

[1] https://wikidp.wiki.opencura.com

[2] https://api.wikimedia.org/wiki/API_reference

[3] https://query.wikidata.org/

[4] https://www.mediawiki.org/wiki/Wikidata_Query_Service/User_Manual

18th International Conference on Digital Preservation
iPRES 2022, Glasgow, Scotland.

DOI: 10.1145/nnnnnnn.nnnnnnn

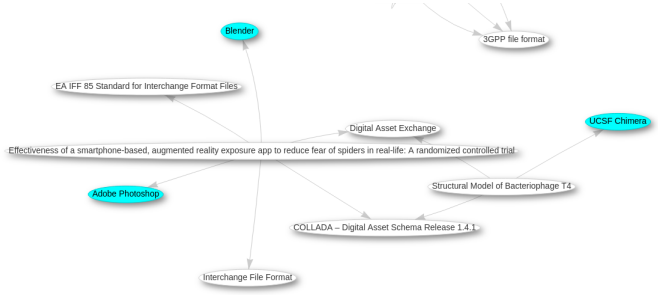

Figure 1: Graph visualization of a SPARQL query illustrating connections in Wikidata between a scholarly publication, a software title (in blue), a file format and a technical specification for that format.

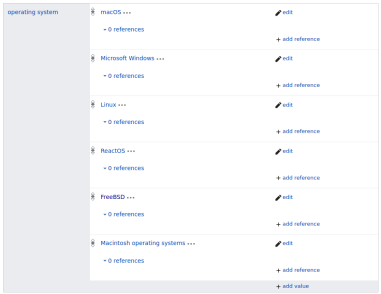

Figure 2: The six operating systems listed on the item for SimulaBeta Q110377565.

Wikidata contains hundreds of thousands of items related to the domain of computing [13]. It contains data about topics from software titles to software development companies, from file formats to operating systems, even computer hardware. As members of the Wikidata community contribute statements to these items, the set of structured data describing the domain of computing becomes more complete. As members of the Wikidata community use more properties to connect items to one another, we can trace context from a scientific article that describes a project that uses a particular piece of software to a general set of information about that software title, to a list of the file formats with which that software title can interact, to a technical specification for the file format itself, as seen in Figure 1. One way to get data out of Wikidata is to write SPARQL queries and run them on the Wikidata Query Service SPARQL endpoint [14].

Not only is the data in Wikidata free for anyone to reuse, the software used to create Wikidata is also available for reuse. The Wikimedia Foundation (WMF) has stewarded the MediaWiki software which is used across the many projects of the WMF. The well-known Just solve the problem project [5] uses Mediawiki software, and the popular Coptr project [6] uses Semantic Mediawiki, which itself is based on Mediawiki.

III. Wikibase

Wikibase is an extension of MediaWiki. MediaWiki is the software used by projects of the Wikimedia Foundation, familiar to most people as the software that powers the different language versions of Wikipedia. Wikibase is the software that enables Wikidata [15]. The German chapter of the Wikimedia Foundation, Wikimedia Deutschland (WMDE), made a docker image available that includes Wikibase in addition to other software [16]. It is available under a free software license allowing anyone to reuse Wikibase to build their own knowledge base.

Anyone can use Wikibase to design a system tailored to their data[7]. People who want to create their own properties to express relationships different from those available in Wikidata can use Wikibase to do so[8]. People who want to structure data that isn't appropriate for inclusion in Wikidata can use Wikibase to do so. The fact that Wikibase includes a SPARQL endpoint means that it is possible to run federated queries across a Wikibase and Wikidata itself which allows people to combine the data in their Wikibase with data from Wikidata. An example of a Wikibase related to digital preservation is the ArtBase created by Rhizome [17]. Additional examples of projects using Wikibase can be found in the Wikibase Registry, itself an instance of Wikibase, that provides details on Wikibase useage[9].

We selected Wikibase for the WikiDP knowledge base because of our familiarity with it from curating data in Wikidata [13], [18], [19]. We wanted to be able to reuse parts of the Wikidata graph in the WikiDP Wikibase. We also wanted to use the Wikibase data model so that we could contribute parts of this data to Wikidata at some point in the future, if the community decides it would be valuable. Wikibase is appropriate for our project because it allowed us to easily make this data available on the web, and it provides a SPARQL endpoint for querying the data.

We decided to create a Wikibase instance for this data because the level of detail required to describe configured software environments involves greater expressivity than is currently possible in Wikidata. We decided that this data model extended too far beyond that of Wikidata, and thus would not be appropriate for inclusion. An example of differences in the level of detail is the way software titles and operating systems are described. In Wikidata, multiple operating systems are listed for a software title to indicate those with which the software is known to be compatible. An example of a Wikidata item with multiple compatible operating systems listed is SimulaBeta (Q110377565) as seen in Figure 2.

In the WikiDP Wikibase we create new items for each

[5] http://fileformats.archiveteam.org/wiki/Statement_of_Project
[6] https://coptr.digipres.org/index.php/Main_Page
[7] Wikibase documentation available here.
[8] There is also a feature known as 'federated properties' which allows Wikibase users to seamlessly reuse properties from Wikidata as described here.
[9] https://wikibase-registry.wmflabs.org/wiki/Main_Page

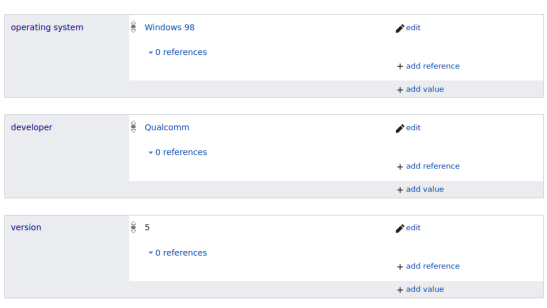

Figure 3: Screenshot of Eudora with Windows 98 listed as compatible operating system in the WikiDP Wikibase.

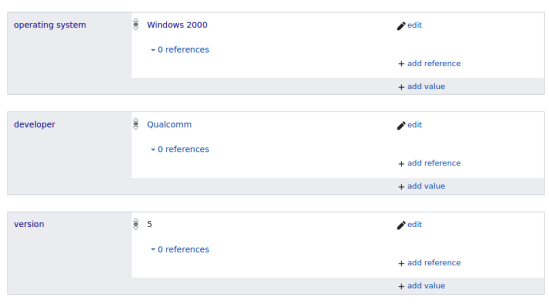

Figure 4: Screenshot of Eudora with Windows 2000 listed as compatible operating system in the WikiDP Wikibase.

software title and operating system combination. This is because we are interested in describing configured environments available in EaaSI and what they contain. Driven by this use case, it is helpful to have each software title and operating system combination modeled as distinct items. This can be seen in Figure 3, showing the software title Eudora with a single value for the operating system property in the WikiDP Wikibase and Figure 4, a distinct item for the software title Eudora with a different operating system listed. As users of EaaSI use pre-configured environments, it is helpful to have different items for each software tile and operating system combination.

Wikibase has worked very well for this use case. All of the data is public, so there are no issues with the data being available on the web. The process of creating items and properties is familiar to editors of Wikidata. Reusing data from Wikidata allowed us to make useful and meaningful connections between the NSRL data, which was previously siloed, with a general-purpose data set describing computing resources.

IV. Wikidata Subsetting

Data published in Wikidata is available under a Creative Commons Zero (CC0) license[10], meaning anyone can reuse any data from Wikidata for any purpose. When creating a new Wikibase, it is sometimes desirable to reuse one or more subsets of Wikidata in the new knowledge base. Creating a subset involves identifying the items and statements about those items you are most interested in and writing a query to extract them from Wikidata. Due to the coverage of items related to the domain of computing, we were able to reuse data from Wikidata to populate our WikiDP Wikibase with structured data. Reusing subsets of Wikidata reduces time needed to source and structure that data. Reusing subsets of Wikidata in Wikibase instances is also convenient because of the fact that they share the same underlying data model.

We used WikidataIntegrator (WDI) to fetch subsets of Wikidata and to populate the WikiDP Wikibase with that data. WDI is a Python library for interacting with data from Wikidata [20]. WDI was created by the Su Lab of Scripps Research Institute and published under an open-source software license via GitHub[11]. WDI can be used to pull data from Wikidata or to populate Wikidata with data. Similarly, WDI can also be used to get data from or write data to a Wikibase.

We created direct mappings to corresponding Wikidata items for several classes in WikiDP. We reused a subset of Wikidata covering human languages, creating items for each language in WikiDP, and creating a mapping back to Wikidata. We added these items so that we could use them to indicate the languages in which the user interfaces of software titles are available. We also reused the file format subset of Wikidata so that we could reuse them in the Wikibase. Each of the file format items also has a statement containing a mapping back to Wikidata.

Maintaining these mappings is useful for writing federated SPARQL queries. A federated SPARQL query requests information from two or more endpoints in a single query. For example, because of the mappings between file format items in the WikiDP Wikibase and their counterparts in Wikidata, we can ask questions about the file formats in the WikiDP Wikibase and also retrieve data from Wikidata in a single query. Figure 5 shows a SPARQL query that asks for file formats in the WikiDP Wikibase that have a mapping to Wikidata, and then uses that mapping to find the equivalent file format items in Wikidata that have been used as a value for the property 'main subject' on scholarly article items in Wikidata. The query allows us to see a list of scholarly articles that describe file formats.

Another example of a federated query between the two systems allows us to retrieve user manual links for certain software titles, as seen in Figure 6. The software in the NSRL collection does not include user manuals for the majority of titles. Users of EaaSI may need to consult the user manual for the software they are using in a given configured environment. Some of the software titles in Wikidata contain links to a copy of their user manual. By combining data from both knowledge bases we can supply user manual links for many of the NSRL software titles. The software titles are from the NSRL collection in the WikiDP Wikibase, but the user manual links

[10] https://creativecommons.org/choose/zero/

[11] https://github.com/SuLab/WikidataIntegrator

Figure 5: Federated query on the WikiDP Wikibase SPARQL endpoint combining data from Wikidata with data from the WikiDP Wikibase. Try it!

Figure 6: Federated SPARQL query on the WikiDP endpoint.

are from Wikidata.

The NSRL collection contains software titles produced by Brøderbund, but does not contain any information about Brøderbund itself. If we consult Wikidata to see what information about Brøderbund has been added, we find a wide range of information. An archival collection related to the company is held by The Strong, as seen in Figure 7.

The Brøderbund item in Wikidata also contains information about a list of Brøderbund products from English Wikipedia as well as the category on English Wikipedia for Brøderbund games, as seen in Figure 8. Additionally, the item provides sitelinks to 21 articles in different language versions of Wikipedia about Brøderbund.

At the bottom of the page of the Wikidata item there are forty-three external identifiers listed. External identifier properties are used in Wikidata to provide links out to where a resource, in this case Brøderbund, is described by other sites. Wikidata has become a hub for storing and managing identifiers for items [21]. Rather than search for Brøderbund using the search options provided by these forty-three systems, this information is now stored in Wikidata, easing discovery. A sample of some of the external identifiers found on the Wikidata item for Brøderbund can be seen in Figure 9. Several national libraries have information about Brøderbund in their collections. Crunchbase, a database of technology companies has information about the corporate profile of Brøderbund. Justia Patents has information about patents filed or held by Brøderbund. General information about Brøderbund from Wikidata can be combined with information from the NSRL that describes specific software titles that Brøderbund developed.

Figure 7: Information about archival collection on the Wikidata item for Brøderbund.

Figure 8: Statements on the Wikidata item for Brøderbund providing information about related information from English Wikipedia.

After mapping items and classes from the WikiDP Wikibase to Wikidata we can contextualize information about the NSRL software within the larger sets of information about developers available in Wikidata. Depending on our use cases or our research needs, we can also quickly identify other resources on the web, like the Media Arts Database or the Justia Patents database, if we are interested in specific types of additional information.

Wikidata subsetting is an effective strategy for populating slices of data into a Wikibase. Establishing a property to store the Wikidata mapping for a corresponding item or property in Wikidata itself is useful for anyone who creates a Wikibase and plans to create mappings to

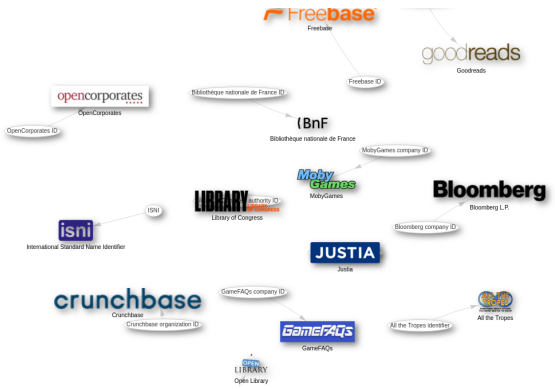

Figure 9: Logos of some organizations that operate repositories for which there is an external identifier related to Brøderbund in Wikidata.

Wikidata. As more Wikibases are created in this way we will see the ecosystem of Wikibases diversify in terms of content and data models. This will supplement Wikidata, and provide flexibility for organizations with specific use cases and modeling needs.

V. National Software Reference Library

The National Software Reference Library (NSRL) is a collection of software and metadata about software created by the National Institute of Standards and Technology (NIST) of the United States. The purpose of the collection is to support research and investigation related to computer forensics [22].

NIST staff created the NSRL by collecting physical copies of software titles across distribution formats. They described the software using a set of metadata properties such as manufacturer, language, compatible operating systems, etc. We compared the inventory of software titles in the NSRL with those described in Wikidata and found only a small area of similarity. NIST donated copies of software titles and associated metadata from the NSRL to Yale University Library as part of the EaaSI program of work. These software titles are being used by EaaSI team members to create a broad range of pre-configured software environments that are available as part of EaaSI.

After reviewing the metadata in the NSRL collection, we designed a set of properties for the WikiDP Wikibase. We considered how we could align certain properties with Wikidata properties. We also considered the needs of the EaaSI system. The final set of properties that we created was influenced by these considerations.

VI. Data Models

We created data models for software titles, software families, file formats, and configured software environments in the WikiDP Wikibase. We use these data models to communicate expectations about data structuring for these different classes of items in the knowledge base.

The EaaSI system provides a catalog of pre-configured software environments for users. These environments are configured by members of the EaaSI team from software available from the NSRL. The class of configured software environment items in the WikiDP Wikibase represents the set of software environments that have been described in the WikiDP Wikibase.

We first created a set of properties inspired by Wikidata. Some examples of these properties are: instance of P1, developer P2, version P3, and file extension P4. Each property also has a mapping to the corresponding Wikidata property as seen in Figure 10. We designed these properties to reflect their equivalent properties in Wikidata so that it would be simple to contribute the data back to Wikidata in the future.

We also created properties to model the NSRL metadata. Some examples are: NSRL manufacturer ID P8, etid P10, etidparent P9, Application ID P11, and NSRL application type P12. These properties reflect the metadata model of the original NSRL corpus.

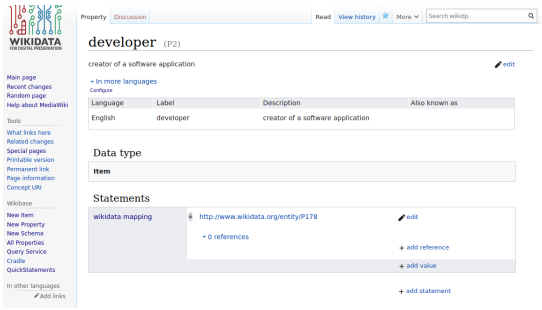

Figure 10: The property for 'developer' in the WikiDP Wikibase with a mapping to the corresponding Wikidata property.

Specific properties we created for EaaSI include: Library of Congress copyright ID P16, base environment, P30, number of disks P38, and Internet Access Required P40. We designed these properties to reflect aspects of how a configured software environment are described.

The Wikibase data model includes references. The references data model makes it possible to reference individual statements. In this way, it is possible to source different statements on the same item to different sources, if needed. It is also possible to provide multiple references per statement. Applying references to each statement ensures that when results are returned via SPARQL, we can quickly identify the source of the information. The reference structure supported by the Wikibase software has been effective for Wikidata [23]. Building on our experiences with the Wikidata system, we work to add references to as many statements as a possible in WikiDP. People who reuse this data will be able to see, per statement, where the data originated and make decisions on whether or not it is relevant for their use case.

VII. Shape Expressions

Shape Expressions (ShEx) is a formal modeling and validation language for RDF data [24]. ShEx is the schema language used in the Schema namespace (namespace E) of Wikidata and other Wikibase instances [25]. ShEx is the language we use to represent our data models. We write schemas in ShExC, the ShEx compact syntax. We publish our schemas in the E namespace of the WikiDP Wikibase. The schemas describe the properties and references that are expected for a class of items as well as their expected values. Schemas are a concise way to communicate data models. People interested in contributing to the WikiDP Wikibase, or reusing data from the WikiDP Wikibase, can consult our schemas to gain understanding of our data models.

Once we have encoded a data model as a schema, we can then use these schemas to validate the entity

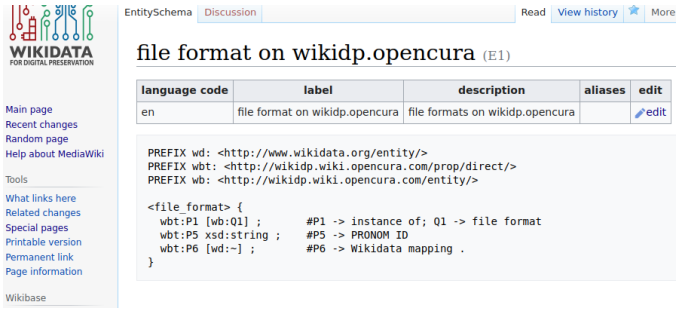

Figure 11: ShEx schema for file formats for the WikiDP Wikibase.

data in our Wikibase. For example, if someone contributes an item describing a configured software environment, they can then validate that item against our schema for 'configured software environment' to test it for conformance. The ability to test entity data for conformance to a schema is useful in the open contribution model of the WikiDP Wikibase. Contributors from different institutional contexts, language backgrounds, and with different use cases for software emulation may want to describe configured environments in the WikiDP Wikibase. As they are becoming familiar with the system, testing the data they contribute for conformance to a schema provides a way to get automated feedback about where the data are not yet conformant, and what types of changes are needed to bring the data into conformance.

The schema for file formats in the WikiDP Wikibase is seen in Figure 11. This schema has a label and description to provide information about the content. Then there are prefix declarations that provide the namespaces from which the properties are derived. There is one shape in this schema and it is called "file format". The file format shape describes three triple patterns. First file formats should all have a statement that they are instances of (P1) file format (Q1). Then they may have a statement that provides their PUID in the form of a string. Lastly, they should have a Wikidata mapping (P6) that provides a Wikidata URI for the corresponding file format in Wikidata.

Writing schemas to describe our data models allows us to communicate how our Wikibase connects to Wikidata itself. This can be useful for people looking to reuse our data, or reuse our data in combination with data from Wikidata. It is also useful for indicating how our Wikibase fits into the network of Wikibases beyond Wikidata.

VIII. Ecosystem of Wikibases

While the breadth of Wikidata content spans many domains, not all data can be accommodated in the knowledge base. The German chapter of the Wikimedia Foundation, Wikimedia Deutschland (WMDE) promotes the concept of an ecosystem of Wikibases [16]. An ecosystem of Wikibases is a network of Wikibase instances each of which supports federated queries with Wikidata itself.

Wikidata was the only Wikibase instance for several years. The Docker image for Wikibase was created by Adam Shoreland and first made available in 2017 [12]. The Wikimedia Foundation has outlined a vision for how interconnected Wikibases will be created for many different uses[13]. The strategy describes how operators of Wikibase instances and developers of related tooling will work in concert to allow people to query multiple resources in order to bring together relevant data.

This ecosystem will encourage groups of people to explore setting up their own Wikibases to serve their own use cases. Some groups may be interested in data that is not appropriate for Wikidata, but can be usefully structured by reusing some properties from Wikidata. Some groups may be interested in creating a set of properties for their data that are not available in Wikidata. Some groups may reuse a subset of Wikidata properties in combination with a set of properties not available in Wikidata. As each Wikibase instance has a SPARQL endpoint that supports federated queries with Wikidata, data can be more easily combined with data from Wikidata.

Both Wikidata itself, as well as the ecosystem of Wikibases, represent the vision of the Semantic Web. "Semantic Web is the idea of having data on the Web defined and linked in a way that it can be used by machines not just for display purposes, but for automation, integration, and reuse of data across various applications" [26]. The Wikidata knowledge base fulfills the requirements outlined for the Semantic Web in that each resource has a unique identifier, is liked to other resources by properties, and that all of the data is machine actionable.

IX. Conclusion

Our work setting up this Wikibase instance and populating it with data has allowed us to interact with the metadata about software titles from the National Software Reference Library (NSRL) in new ways. The data is available on the web and can be searched via the search box in the interface as well as via SPARQL. The data can also now be combined with data from Wikidata.

Creating a Wikibase instance for a specific purpose allows you to establish your own set of properties. This is helpful if you need to represent data models that are not yet represented in Wikidata, or unlikely to be appropriate for Wikidata. For example, there are dozens of properties related to software in Wikidata, but there are many properties important to the data model for configured software environments that are not yet in Wikidata.

The SPARQL endpoint of the WikiDP Wikibase enables federated queries with other SPARQL endpoints.

[12] https://addshore.com/2017/12/wikibase-docker-images/
[13] https://meta.wikimedia.org/wiki/LinkedOpenData/Strategy2021/Wikibase

This SPARQL endpoint allows us to leverage the benefits of combining multiple RDF data sets to ask questions of our data in the context of additional data. Effectively, this mean we can ask questions of multiple databases with a single query.

We have contextualized the software described in the NSRL by strategically mapping parts of its data model to Wikidata. This means that we can now ask questions of the NSRL data that previously were impossible. For example, rather than asking about connections between a software developer and software titles that involve querying strings that represent entities, we can now ask questions that extend to the geographic locations of the headquarters locations of those software developers. Or we can ask questions that extend to the scholarly literature that describes research involving those software titles. Mapping the NSRL data to Wikidata yields URIs for the entities in the Semantic Web for those organizations. With those URIs we can tap into all of the structured data describing them that has been added to Wikidata.

As more people create Wikibases and populate them with relevant data sets, the ecosystem of repositories of structured data connected to Wikidata will grow and diversify. More people will map previously-siloed data sets to Wikidata, thus creating pathways to the linked open data (LOD) cloud [27]. These connections will unlock access to additional information sources that increase the value of these data sets. In this way, we can transform databases and information systems that were previously islands of data into linked clusters in the LOD cloud.

As an early member of the ecosystem of Wikibases, we expect that many additional Wikibases will be created in future years. As more organizations identify knowledge graphs they would like to have access to on the web that extend beyond the boundaries of Wikidata, many will decide to manage their own Wikibase instances.

Acknowledgments

We would like to thank Adam Shoreland for creating wbstack and Rhizome for funding wbstack. Thank you to the Su Lab at Scripps Research Institute for creating WikidataIntegrator and making it available under an open license. Thank you to Andra Waagmeester for maintaining and creating new features for WikidataIntegrator. We would like to thank the Andrew W. Mellon Foundation and the Alfred P. Sloan Foundation for generously supporting the EaaSI program of work.

References

[1] L.-A. Kaffee, K. M. Endris, and E. Simperl, "When humans and machines collaborate: Cross-lingual label editing in wikidata," in *Proceedings of the 15th International Symposium on Open Collaboration*, 2019, pp. 1–9.

[2] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific american*, vol. 284, no. 5, pp. 34–43, 2001.

[3] J. Hunter and S. Choudhury, "A semi-automated digital preservation system based on semantic web services," in *Proceedings of the 4th ACM/IEEE-CS joint conference on digital libraries*, 2004, pp. 269–278.

[4] Y. Marketakis and Y. Tzitzikas, "Dependency management for digital preservation using semantic web technologies," *International Journal on Digital Libraries*, vol. 10, no. 4, pp. 159–177, 2009.

[5] D. Tarrant, S. Hitchcock, and L. Carr, "Where the semantic web and web 2.0 meet format risk management: P2 registry," *International Journal of Digital Curation*, vol. 6, no. 1, pp. 165–182, 2011.

[6] C. Schlieder, "Digital heritage: Semantic challenges of long-term preservation," *Semantic Web*, vol. 1, no. 1-2, pp. 143–147, 2010.

[7] J. Hunter and S. Choudhury, "Panic: An integrated approach to the preservation of composite digital objects using semantic web services," *International Journal on Digital Libraries*, vol. 6, no. 2, pp. 174–183, 2006.

[8] E. Cochrane, K. Rechert, S. Anderson, J. Meyerson, and E. Gates, "Towards a universal virtual interactor (uvi) for digital objects," 2019. [Online]. Available: https://osf.io/xdehm/download.

[9] D. Vrandečić, "Wikidata: A new platform for collaborative data collection," in *Proceedings of the 21st International Conference Companion on World Wide Web*, ACM, 2012, pp. 1063–1064.

[10] F. Erxleben, M. Günther, M. Krötzsch, J. Mendez, and D. Vrandečić, "Introducing wikidata to the linked data web," in *The Semantic Web–ISWC 2014*, Springer, 2014, pp. 50–65.

[11] S. Harris, A. Seaborne, and E. Prud'hommeaux, *Sparql 1.1 query language, w3c recommendation*, 2013. [Online]. Available: https://www.w3.org/TR/2013/REC-sparql11-query-20130321/.

[12] F. Manola and E. Miller, *Resource description framework: Primer*, 2004. [Online]. Available: https://www.w3.org/TR/2004/REC-rdf-primer-20040210.

[13] K. Thornton, E. Cochrane, T. Ledoux, B. Caron, and C. Wilson, "Modeling the domain of digital preservation in wikidata," *iPRES 2017: 14th International Conference on Digital Preservation*, 2017.

[14] S. Malyshev, M. Krötzsch, L. González, J. Gonsior, and A. Bielefeldt, "Getting the most out of wikidata: Semantic technology usage in wikipedia's knowledge graph," in *International Semantic Web Conference*, Springer, 2018, pp. 376–394.

[15] L. Zhou, C. Shimizu, P. Hitzler, *et al.*, "The enslaved dataset: A real-world complex ontology alignment benchmark using wikibase," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 3197–3204.

[16] D. Diefenbach, M. D. Wilde, and S. Alipio, "Wikibase as an infrastructure for knowledge graphs: The eu knowledge graph," in *International Semantic Web Conference*, Springer, 2021, pp. 631–647.

[17] L. Rossenova, D. Espenschied, and K. de Wild, "Provenance for internet art using the w3c prov data model," in *Proceedings of the 16th International Conference on Digital Preservation*, 2019. [Online]. Available: https://osf.io/6xd4g/download.

[18] K. Thornton and K. Seals-Nutt, "Getting digital preservation data out of wikidata," in *Proceedings of the 16th International Conference on Digital Preservation*, 2019. [Online]. Available: https://osf.io/guj3p/#!.

[19] K. Thornton, K. Seals-Nutt, E. Cochrane, and C. Wilson, *Wikidata for digital preservation*, 2018. [Online]. Available: 10.5281/zenodo.1214319.

[20] A. Waagmeester, G. Stupp, S. Burgstaller-Muehlbacher, *et al.*, "Wikidata as a knowledge graph for the life sciences," *Elife*, vol. 9, e52614, 2020. [Online]. Available: https://doi.org/10.7554/ELIFE.52614.

[21] J. Neubert, "Wikidata as a linking hub for knowledge organization systems? integrating an authority mapping into wikidata and learning lessons for KOS mappings," in *Proceedings of the 17th European Networked Knowledge Organization Systems Workshop co-located with the 21st International Conference on Theory and Practice of Digital Libraries 2017 (TPDL 2017), Thessaloniki, Greece, September 21st, 2017.*, 2017, pp. 14–25. [Online]. Available: http://ceur-ws.org/Vol-1937/paper2.pdf.

[22] S. Mead, "Unique file identification in the national software reference library," *Digital Investigation*, vol. 3, no. 3, pp. 138–150, 2006.

[23] A. Piscopo, L.-A. Kaffee, C. Phethean, and E. Simperl, "Provenance information in a collaborative knowledge graph: An evaluation of wikidata external references," in *International semantic web conference*, Springer, 2017, pp. 542–558.

[24] I. Boneva, J. E. L. Gayo, S. Hym, E. G. Prud'hommeaux, H. R. Solbrig, and S. Staworko, "Validating RDF with shape expressions," *CoRR*, vol. abs/1404.1270, 2014. [Online]. Available: http://arxiv.org/abs/1404.1270.

[25] K. Thornton, H. Solbrig, G. S. Stupp, *et al.*, "Using shape expressions (shex) to share rdf data models and to guide curation with rigorous validation," in *European Semantic Web Conference*, Springer, 2019, pp. 606–620.

[26] I. Cruz, S. Decker, J. Euzenat, and D. McGuinness, *The emerging semantic web*.

[27] D. Abián, F Guerra, J Martínez-Romanos, and R. Trillo-Lado, "Wikidata and dbpedia: A comparative study," in *Semanitic Keyword-based Search on Structured Data Sources*, Springer, 2017, pp. 142–154.