Wikidata: A Magic Portal for Siegfried and Roy

Towards a resolution for open registries and file format identification

Ross Spencer
Ravensburger AG
Germany
ross.spencer@ravensburger.de
0000-0002-5144-9794

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

Richard Lehane
International Atomic Energy Agency
Austria
r.lehane@iaea.org
0000-0002-9507-4816

Euan Cochrane
Yale University Library
United States
euan.cochrane@yale.edu
0000-0001-9772-9743

Abstract – A project team was established in late 2019 to create a Siegfried/Wikidata integration. The project was established to make Yale University Library's work curating digital preservation data in Wikidata actionable; that is, enable it to be consumed by the Siegfried utility for the purpose of file format identification. Once accessible by such a tool, file-system data can be linked directly to Wikidata records and thus the breadth and depth of the Wikidata dataset. The combination opens up possibilities around the EaaSI work also being developed with Wikidata in mind. The Siegfried/Wikidata collaboration was an intentional one that would have been difficult to make happen through normal circumstances; be that geographic or organisational dispersal, opportunity, or cost. The project was seen as a worthwhile endeavour to pursue independently of these restrictions and other projects may in the future need to resort to non-traditional approaches to achieve their aim. The first iteration of this work successfully went live in October 2020 and for it to continue to succeed there is a call to arms from those in the digital preservation and Wikidata communities to help push it further. Work in the pipeline is believed will continue to demonstrate the worthwhile nature of today's commitment.

Keywords – Wikidata, WikiDP, Format-Identification, Siegfried, Collaboration.

Conference Topics – Enhancing the Collaboration; Building the Capacity & Capability.

I. Introduction

In 2020 an agile project was established to undertake the integration of Wikidata with the file format identification tool Siegfried. The integration would allow Siegfried to access Wikidata file format records to extend Siegfried's identification capabilities and enable it to deliver metadata as it is available in Wikidata. The project team consisted of Euan Cochran (product owner - Yale University Library (YUL)), Kat Thornton (Wikidata subject matter expert (SME) - (YUL)), Richard Lehane (maintainer of Siegfried), and Ross Spencer (format identification (SME) and developer).

The Siegfried/Wikidata integration seeks to develop Wikidata as a digital preservation technical registry enabling different workflows through the many combinations of data exposed through it. Format identification is a small but early part of many digital preservation workflows that can be expanded on to provide, for example, rendering of legacy content in the GLAM (Galleries, Libraries, Archives, Museums) sector through emulation services. With YUL's work on Emulation as a Service (EaaSI) [1] the project was a natural fit.

The project successfully integrated the two services enabling file format identification and delivery of rich metadata using the Wikidata Query Service[2]. The project also delivered a number of ancillary technologies enhancing the Golang[3][4] ecosystem around linked open data. Work is also underway looking at how to further enhance Wikidata for container and multi-part file format identification, as well as understanding how best to integrate and make use of the many potential sources of file format information in the service, e.g. making use of recently added TrID[5] format-signatures.

With the first iteration of this work available to users of the digital preservation and Wikidata communities there are components of work that still need to be com-

[1] Information about EaaSI on the Software Preservation Network: https://perma.cc/BTG8-MZ25

[2] Wikidata Query Service endpoint: https://perma.cc/PJQ6-ZUZM

[3] Golang.org: https://perma.cc/L49S-YPLR

[4] Golang is the programming language of choice for Siegfried. It provides support for cross-platform development and provides built-in support for a number of development activities such as documentation, unit-testing, and benchmarking

[5] TrID format identification utility homepage: https://perma.cc/R3WT-383D

17th International Conference on Digital Preservation
iPRES 2021, Beijing, China.

DOI: 10.1145/nnnnnnn.nnnnnnn

pleted to help enable engagement across both.

The remainder of this paper will review the collaboration that enabled this project to happen. Its motivations; future work; as well as the next steps in the respective communities required to see the Siegfried/Wikidata integration continue to be a success.

II. Motivation

If successful, the integration could turn Wikidata into a dynamic, open repository, of "actionable" file format information - linking a utility like Siegfried to the web and a wealth of potential other resources supporting users' digital preservation needs.

The integration provides a ready-made repository and collaboration platform for storing file format information that anyone familiar with the way of working with services like Wikipedia can work with; and those that are not as familiar have the potential to learn.

Wikidata is also a platform that can be hosted by anyone, such an integration could also expand the potential for federated information around the subject of digital preservation to be shared and linked through Siegfried.

Given those ingredients it was felt that it was an important avenue of investigation to follow through with, which could further solidify the vision and efforts of the team at Yale University Library over the past half-decade or so.

III. Format ID

A. *Introduction to format identification*

File format identification refers specifically to the ability to identify file formats based on consistent patterns (magic numbers) in their relative different byte-sequences. So, for, example, old-school Microsoft Office file formats would begin with the hexadecimal numbers: D0CF11E0 which is the magic number of a file format called OLE2 (Object Linking and Embedding 2.0). Magic numbers are more commonly known as file format signatures and the tools: DROID (Digital Record and Object Identification), FIDO (Format Identification for Digital Objects), and Siegfried, are all best-of-breed utilities for working with them.

Each tool is capable of interpreting, not just magic numbers, but usually, a form of regular expression syntax (both DROID and FIDO have their own) which provides a language to describe so-called file format signatures. As with most languages, the DROID or FIDO regular expressions can be pieced together in complex ways to create something both expressive, and able to capture many more possibilities in a single definition.

Where Siegfried differs from the other two utilities is that it was conceived with the idea of being able to consume, not just DROID compatible (PRONOM) signatures, but other forms of file format signature that might exist. To date, it is possible to use PRONOM standard signatures, PRONOM container signatures, and MIME-Info/Tika signatures[6]. It also provides mechanisms for working with filenames/file extensions.

B. *Siegfried and extending Siegfried*

Siegfried encapsulates two components, Roy and Siegfried. Roy is the component that enables the harvest of signature datasets and saves them to a signature file. Siegfried is the identification component and it consumes this file, scans digital objects, and returns the identification results to the user.

Siegfried's ability to work with multiple types of file format signature is delivered by the way it abstracts many of the components in the underlying source code. Developing standard interfaces for storing signatures (identifiers) connecting to the identifier engines (matchers) or returning results to Siegfried (recorders), no matter what the underlying implementation of a matcher may look like.

Including a method for displaying results (identification), Siegfried is split into four interfaces:

1. the Identifier interface. This represents a unit of signatures that can produce a format identification (e.g. a PRONOM release is an Identifier). A single signature file can encode one or more Identifiers.
2. the Matcher interface. This represents a different method of identification (e.g. file name, text characterisation, byte pattern matching). Depending on the Identifiers used, a signature file will have one or more Matchers.
3. the Recorder interface. Recorders are spawned by Identifiers when Siegfried runs against a file (1:1). They are mutable structures that record hits from the Matchers and then report the Identification when the scan ends. They apply the priority or other rules by which an Identifier determines the correct Identification(s) based on hits from the Matchers.
4. the Identification interface. This is the output that a Recorder produces having received signals from the Matchers. E.g. a PUID (fmt/1).

So Siegfried is eminently extensible. Through that capability it could already connect to PRONOM[7], the Library of Congress Digital Format Descriptions[8], or be supplied with either Tika or Freedesktop's MIMEInfo database for identification[9]. As well as combine each of

[6] Generally speaking, Siegfried could be extended to use FIDO style sequences, but at the time of writing, FIDO mostly consumes the PRONOM registry and converts it to a standardized regular expression syntax, that is, one compatible with standard programming libraries such as those available in the Python programming language that FIDO is built upon.

[7] The National Archives, UK - PRONOM: https://perma.cc/FXB6-DWG9

[8] Library of Congress Digital format Descriptions: https://perma.cc/66AX-AT9A

[9] Freedesktop MIMEInfo definitions: https://perma.cc/MN4J-BKWR

those into different combinations of identifiers to potentially generate a more complete coverage of identification results.

The Library of Congress identifier implementation was used as a reference point for extending Siegfried. It was one of the more recent identifiers added to the tool and one of the easiest to comprehend. The Library of Congress identifier encapsulates the parsing of file format definitions and the return of results to the end-user. It is somewhat simpler than other identifiers as it focuses on a single style of binary matcher.

Extending Siegfried's capability meant first extending Roy, adding:

- An ability to connect to the Wikidata Query Service.
- Construction of a suitable SPARQL query for the retrieval of file format data and signatures from the service[10].
- Parsing the SPARQL results into a format that could be persisted by Siegfried into a signature file.
- Association of signatures with their appropriate matchers, e.g. PRONOM based matching, or something else, e.g. Container.
- An ability to inspect the signature file and return information to the caller about how matches have been constructed.

And then extending Siegfried adding:

- An ability to consume the signature file when the command is run.

At this point Siegfried's scanning capability handles the matching of file formats to matching signatures, after which:

- Siegfried needs to display the results of an identification as they are returned. For Wikidata this means making sure that the information being added that provides extra value to the caller is displayed and correctly formatted (additional provenance about the source of a signature in Wikidata was considered an important addition by this effort).

The results can be seen in Table 1.

IV. Wikidata

The Wikidata community began building a knowledge base of structured data in 2012 [1]. This cross-domain knowledge base contains data about topics ranging from food, to computing, to human genes [2]. The Wikidata data model consists of items and properties. Every statement in the knowledge base can be referenced to sources. A full version history is available for the knowledge base.

Data in Wikidata is available under the Creative Commons Zero license, meaning that anyone is free to reuse the data for any purpose. Data may be reused from Wikidata via an application programming interface (API), a public SPARQL endpoint, or by downloading data dumps [11].

There are currently more than twenty-five thousand active editors in the Wikidata community [3]. As of 2021, there are more than ninety-three million items in the Wikidata knowledge base. The Wikidata community is growing steadily. Digital Preservationists wanting to reuse data from Wikidata will be interested to know that the domain of computing is a topic of interest for many Wikidatans. The number of items related to file formats has been growing steadily over the past eight years [4].

A. *Format data in Wikidata*

People in the digital preservation community have been curating data related to file formats in the Wikidata knowledge base for several years [4]–[6]. The Wikidata data model consists of items and properties. Every item in Wikidata has a unique identifier called a Qid. In Figure 1, two properties in use on this item for JPEG 1.02 are shown. In this example the property P31 "instance of" has a value of "file format" which is an item that represents the class of file formats.

Wikidatans connect existing resources related to file formats with Wikidata through the use of external identifier properties. The Wikidata community created property P2748 "PRONOM file format identifier" in April of 2016. In October of 2016, they created P3266 "Library of Congress Format Description Document ID". In December of 2016, they created P3381 "File Format Wiki page ID". These properties are evidence of the community's interest in curating data related to file formats. As of April 2021, there are 13,053 items in Wikidata for file formats. Of these items, 9,411 contain statements about their signatures.

File format signature data is represented in Wikidata by a property and three qualifiers. Qualifiers are properties that can be used to refine the scope of a property. Property P4152 "file format identification pattern" is used to store the signature as a string of characters. Property P3294 "encoding" is used as a qualifier to indicate the character encoding of the signature value. Property P4153 "offset" is used as a qualifier to express the number of bytes before the signature is found in the file. Property P2210 "relative to" is used as a qualifier to indicate whether the offset is from the beginning of the file or the end of the file. The use of these properties and qualifiers on a file format item from Wikidata is depicted

[10] SPARQL in Siegfried 1.9.1 to retrieve file format records from Wikidata: https://perma.cc/GLD8-EZWJ

[11] Accessing information from Wikidata: https://perma.cc/A3YL-ZUMX

```
---
siegfried   : 1.9.1
scandate    : 2021-05-19T19:54:24+02:00
signature   : default.sig
created     : 2021-05-19T19:52:55+02:00
identifiers :
  - name    : 'wikidata'
    details : 'wikidata-definitions-1.0.0 (2021-05-19)'
---
filename : 'skeleton-test-suite/fmt-279-signature-id-295.flac'
filesize : 2600
modified : 2020-11-15T15:36:16+01:00
errors   :
matches  :
  - ns      : 'wikidata'
    id      : 'Q27881556'
    format  : 'Free Lossless Audio Codec'
    URI     : 'http://www.wikidata.org/entity/Q27881556'
    mime    : 'audio/x-ogg; audio/x-flac; audio/flac'
    basis   : 'byte match at 0, 8 (Gary Kessler's Signature Table (source: 2017-08-08))'
    warning :
```

Table 1: Example Siegfried result including Wikidata

in Figure 2.

Kat created a schema in Wikidata's schema namespace to represent the data model for file format identification patterns. The schema is available as E237[12] and can be seen in Figure 3. The schema is written in the Shape Expressions (ShEx) language and can be used to validate instance data from Wikidata [7]. In this way we can identify which file format items are structured in conformance with this schema and which are not. Sharing this data model via the Wikidata schema namespace allows others in the community to view, edit, discuss and reuse this model.

People in the digital preservation community previously have designed and created technical registries of format information [8]–[10]. While these efforts were led by members of the digital preservation community, Wikidata is led by a chapter of the Wikimedia Foundation, Wikimedia Deutschland. Although the mission and vision of Wikidata are broader than providing infrastructure for a technical registry, it is possible to meet the needs of the digital preservation community by storing format information in Wikidata [6]. Data in Wikidata is FAIR data [11]. The visibility of Wikimedia projects on the web increases the findability of this data beyond that of a website dedicated to format information.

B. Other format adjacent information in Wikidata

A variety of properties can be used to connect formats to other items in Wikidata. For example, many software items have statements using properties P1072 "readable file format" and P1073 "writable file format" connecting formats to software items. Many technical specifications also have items in Wikidata, and can be connected to the formats they describe. Scholarly articles that have been written about formats may be linked to the format through the use of property P921 "main subject". In this way, people who are interested in formats can follow links to related resources and documentation for further information.

For the EaaSI program of work, the connections between software items and relevant format items are an important source of information. The EaaSI program of work provides a broad range of different configured emulated environments. Determining the correct subset of configured emulated environments to present to the user requires information about compatible formats per software title.

V. Collaboration

Collaboration was central to this effort, and Siegfried, as has been established, provided a perfect vehicle to expose the data in Wikidata and make it actionable. That being said, undertaking the development work required to make that possible is like taking a second job. At the conclusion of the first iteration, which amounted to seven sprints in total between February and November, 345 development hours were recorded against the project, another 10-40 hours were then spent on project management.

The project was established by Yale University Library, reaching out to Richard Lehane in October 2019. With Ross Spencer's proximity to Vienna at the time and looking to expand his development horizons he was approached and able to agree to take on project coordination and development tasks.

[12] Data model for file format identification patterns in Wikidata: https://perma.cc/V6PG-8C87

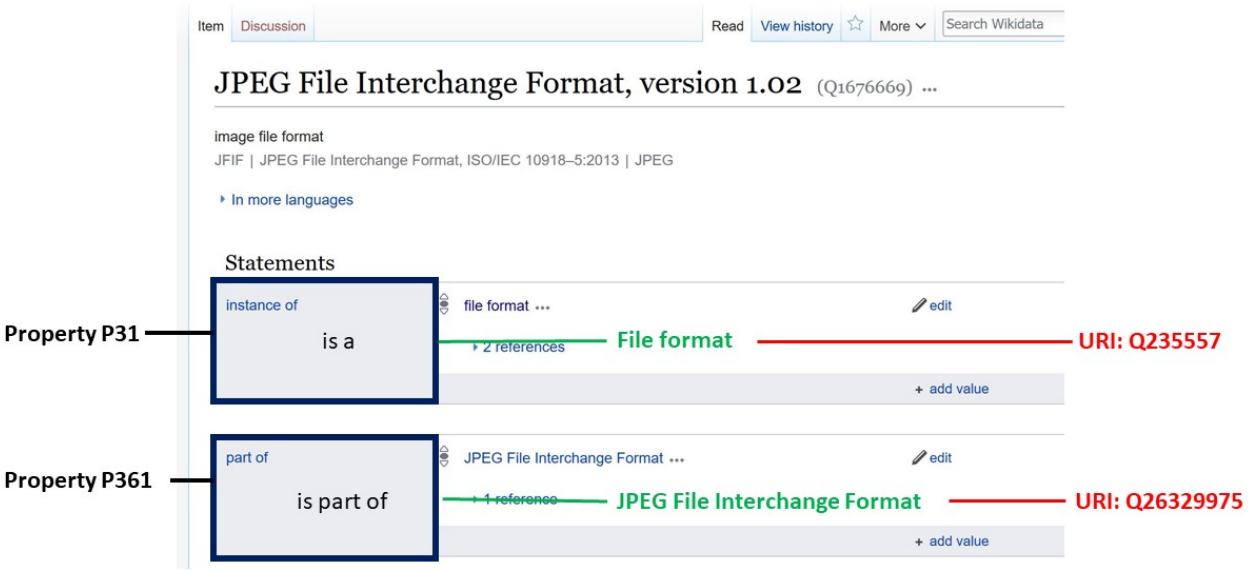

Figure 1: Screenshot of the Wikidata item for JPEG 1.02

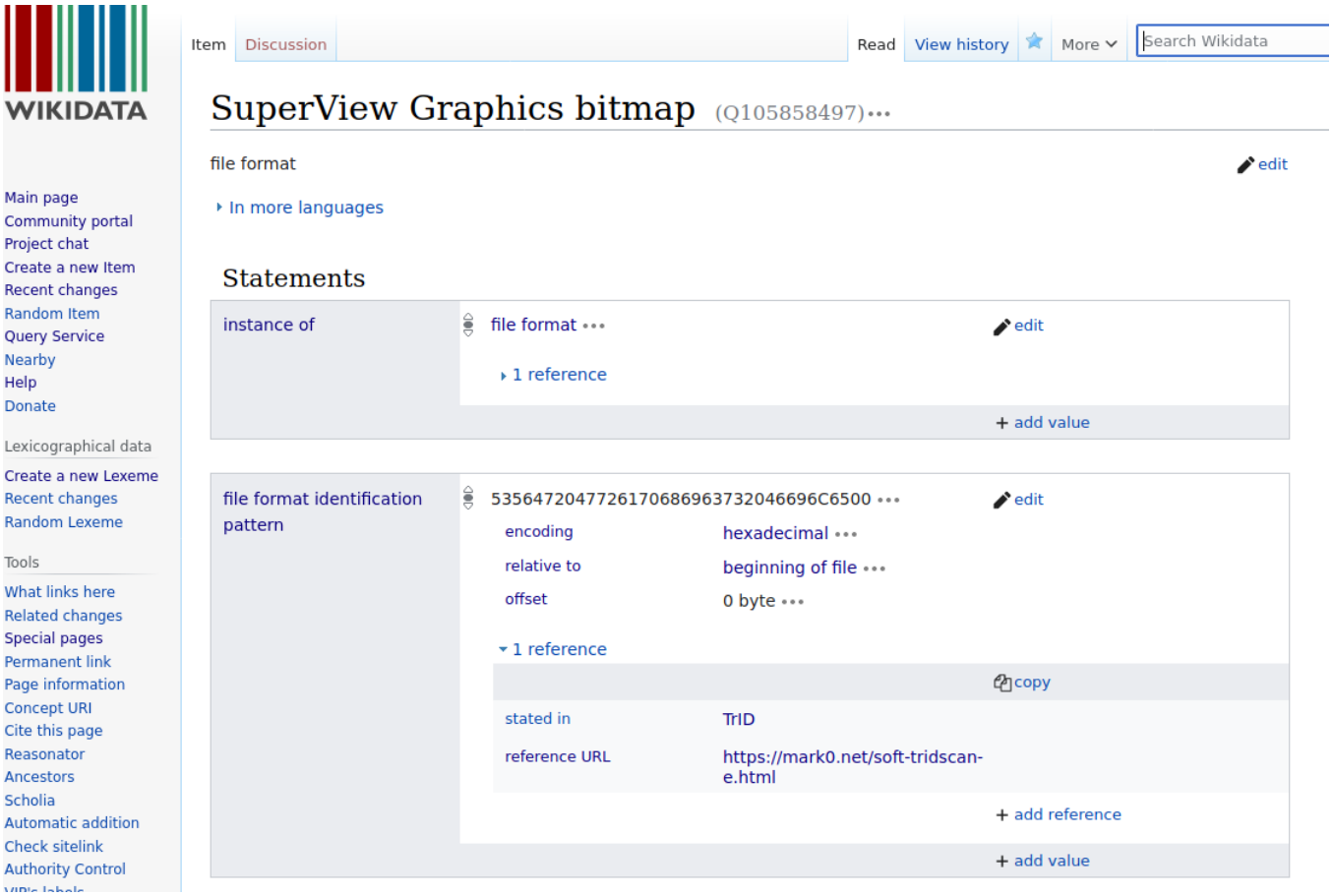

Figure 2: Screenshot of the Wikidata item for SuperView Graphics bitmap format.

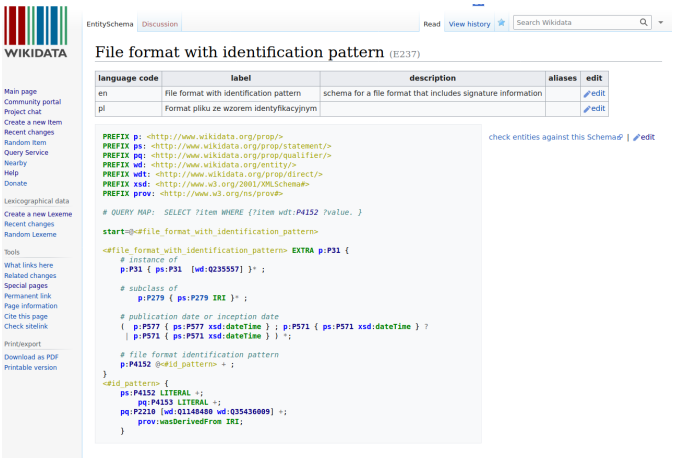

Figure 3: Screenshot of the Wikidata schema for file format with identification pattern E237.

After initial meetings to establish the bounds of the effort and to understand the work on the horizon from other projects, namely, understanding the state of play with PRONOM and introducing the work to the team at The National Archives, UK - the project began in earnest in February 2020.

The first project plan described a total of four sprints over the course of 100 hours to implement a Wikidata identifier in Siegfried plus the other implied efforts around that. The project took an agile approach and as the development requirements became better understood and reshaped, the sprints were able to continue, each becoming more specific in focus until the release of Siegfried 1.9.x.

Sprint meetings were documented and thus provide an important record that will remain important to understanding the direction of the work to continue.

During the entire period, and continuing to this day (an extended Sprint 8), the team members would dedicate time to reviewing documents and proposals as well as attending and contributing to team meetings.

In many regards, the project operated as it might in a vendor-supplier relationship. It was felt that this was important to maintain the quality of the outcome. While talks initially took place to understand if a contract could be established to undertake the work, it was clear that the cost in terms of time could extend well beyond the initial estimates, and while they may not become prohibitive costs, there remained a freedom in not having to take on that burden which is felt helped with the work that was being done.

A. *Reflecting on the collaboration*

The digital preservation community has proven flexible and loose enough over the years to foster collaboration. Like projects such FFMprovisr[13] [14], Just Solve It - File Formats[15], COPTR[16], or bwFLA[17] in its early days and even EaaSI[18] today - the effort needed to make this development happen was necessarily cross-institution and cross discipline. It also had a clear trigger and vision from the members at Yale University Library.

Citing those projects above, the impact of what can be done with enough motivation can be seen across the community. That being said, the community model aside, there is a limit to what can be achieved with community labor alone. The next stages for the Siegfried/Wikidata collaboration is to grow it further, seek buy-in, and institutional support that values it, and can thus generate more value from it.

The investment needed so far to create a stable, working release of Siegfried on-top of the open Wikidata registry is well documented in this paper and the project record - but it is also still far from being complete. As will be discussed, only one other piece of the infrastructure has been put in place. That infrastructure now needs to grow into an ecosystem.

VI. Outcome

The project delivered a first iteration of the Siegfried/Wikidata integration - Siegfried 1.9 was released in October 2020. At the time of writing there are over 13,000 file format records in Wikidata with information that has the potential to be used. Nearly 9000 records have signatures that can be readily consumed by Siegfried to perform the type of pattern matching against file formats described above that lead to a file format identification.

Less tangible than the technology, the collaboration acted as a platform to on-board another developer to

[13] Proposal for FFMProvisr at AMIA hack-day 2015: https://perma.cc/S689-VFNX

[14] Sustainability Through Community - ffmprovisr and the Case for Collaborative Knowledge Transfer (iPRES 2019): https://perma.cc/S4PL-KJB5

[15] Just Solve It - Statement of Project: https://perma.cc/2D5B-Y6RG

[16] Creating a Community Owned Digital Preservation Tool Registry: https://perma.cc/X7RK-CPMP

[17] Software curation as a digital preservation service: https://perma.cc/B76B-6JS4

[18] Making things EaaSIer - Overview from EaaSIs PI: https://perma.cc/P4VM-77LK

Siegfried. While already a long-time contributor to the project, Ross learned about a big part of the Siegfried code-base that will lend itself well to helping support the project in future. Throughout the project's lifespan, every member of the team was exposed to new aspects of its various components, whether that be through Kat's long history and knowledge of Wikidata, or Richard's appreciation and comprehension around the ways in which file format identification can evolve; colleagues have all been able to take something away that is new to them.

Beyond the Wikidata integration, that newly found knowledge enabled the additional development of a small demonstration of how ISO disk-image extraction could be integrated into Siegfried, and as part of that, a more concrete implementation of an "archive format selector" (ZIP, TAR, GZIP, WARC, ARC) allowing users to elect which "archive" file formats to scan the contents of at run-time[19]. Various other small fixes were added through this new collaboration along the way.

Documentation was enhanced on the Siegfried Wiki[20]. The code for the integration was also documented so as to be available via Golang's documentation infrastructure as well[21]. Command line-flags were added to the tool that describe clearly which features can be used.

Two pieces of the puzzle that were developed and are described below, 'spargo' and Siegfried's static analysis 'linting' of Wikidata are central to the development of new capabilities. These features are likely to be further developed in the future.

A. *Spargo*

Spargo[22] is developed and maintained separately from the primary Siegfried repositories at the time of writing. It was one of the first components written for the Siegfried/Wikidata integration. It was developed to interact with the Wikidata SPARQL endpoint. While, for now, a third-party dependency from Siegfried's perspective, it is still in close proximity to the project which should make it easy to maintain and ensure that its features do not skew relative to Siegfried over a longer-period of time. Spargo is a library and can work with the results of any SPARQL endpoint and so also enhances the Golang and linked open data ecosystem for developers requiring that capability.

B. *Linting*

Linting is a static analysis technique for information (e.g. programming language source code, or other dataset) that can describe errors or potential errors in a source that a user can then 'fix-up'. A spell-check is analogous to a linter. In software a linter might identify problems in code against a stylistic standard, so for example, identify when a developer is using "camelCase" in Python variable naming where "snake_case" is more generally recommended[23].

From Wikidata's perspective there is a lot of information to work with. While the data has been carefully curated, it does not all work out of the box with a tool such as Siegfried. Just as with other data sets, information can be missing or incomplete. Information might also be entirely incompatible, or need converting to something that is compatible. Given the number of records in the system, it is nearly impossible for a person to work on those alone to correct them without some form of automation pointing out the potential gotchas.

Linting for Wikidata is built into Roy's handling of the Wikidata signatures when building a new identifier. It can report on missing or incompatible information. It separates messages into warnings and errors based on whether they can still be used by the tool. If they can still be used by Siegfried then the information will make it through to the new identifier.

Linting is a new concept to Roy and is provided through the `‘-wikidatadebug‘` flag. The output can be accessed by Signature, and Wikidata developers, and can be used to improve the consistency of the underlying information.

VII. Future work

The work continues today, taking the time to consider ways that the utility can be built-out, and the community can be established.

A. *In the pipeline…*

There are various pieces of this work still in development, these are discussed briefly below.

1. *More linting*

Linting provides a means to an end in Siegfried that means that it is possible for the tool itself to wrangle information in Wikidata into a consistent state for reuse, while also offering the developer the opportunity to then go and fix the state in Wikidata itself.

If linting does sit within the scope of the tool, then from a development perspective it can benefit from refactoring so that it can be further de-coupled from the Wikidata identifier in Roy - perhaps factoring it out into its own library that only returns information Roy absolutely needs to work with.

If such static analysis doesn't sit within the scope of the tool then alternatives need to be considered such as a utility that is in support of submitting format identification data to Wikidata, or a schema within Wikidata

[19] Archive Selector Feature in Siegfried: https://perma.cc/V9YU-XCQ6

[20] Wikidata identifier information on the Siegfried Wiki: https://perma.cc/6DJ7-ZVQG

[21] Golang documentation for the Wikidata identifier: https://perma.cc/VVK5-4FH6

[22] spargo Golang package: https://perma.cc/7EGZ-B8XY

[23] Function and Variable names in Python's PEP8 standard: https://perma.cc/HU5Z-WJ2B

itself, for example, using ShEx, described above, to validate the shape of file format graphs that include signature information that can be used by Siegfried to ensure that for each namespace that Wikidata includes, e.g. PRONOM, TrID, Kessler's - the information is correct and exists with the correct cardinalities and restrictions, e.g. making sure that signatures only have one beginning of file sequence, or that beginning of file sequences do not overlap. The same for variable or end of file sequences.

Beyond changing how linting is implemented - to improve the experience of working with Wikidata for extracting format identification information, or reworking it to make it usable - more thinking also needs to go into how the information is returned from a static analysis tool for Wikidata and what it should mean to the caller. Linting will normally return warnings or errors to the user, but do there need to be other levels of notification? Do the messages reflect what changes need to be made to the source well-enough? There are certainly aspects of the user-experience around this feature that require further investigation.

2. *Provenance*

It is not yet possible to easily retrieve a Wikidata record's edit history, that is, the edit history for each triple that makes up a record. Multiple triples are used for a format identification and that information can be changed very easily through the Wikidata infrastructure[24]. While there are initiatives to create a Wikidata history[12], they are also stale, or not ready for use in production[25]

In reaction to this, an alternative has been found to provide a stop-gap for Digital Preservationists to access the provenance of file format identification to understand changes between versions that might break existing preservation workflows.

wikiprov[26] is the first library extending spargo. wikiprov wraps the SPARQL work of spargo and creates a complimentary data structure that also contains Wikibase revision history generated from the WikiMedia API, as well as WikiMedia style permalink[27] (permanent link) data, for the records that make up an identifier at a given point in time, i.e. at the time the Wikidata signatures were harvested, which has the potential to be different between two separate downloads [28].

wikiprov's information will be accessed and delivered to the user via Siegfried, e.g. through its results reporting, or Roy inspect[29] functions.

[24] It is also being carefully curated, monitored, and discussed, before information simply changes

[25] Pellisier's Wikidata History Service page on Wikidata which at the time of writing was last updated December 2020 but shows a 5xx error on the service itself: https://perma.cc/Q653-LHWQ

[26] wikiprov Golang package https://perma.cc/A7RW-QJA2

[27] Wikimedia page describing a permalink: https://perma.cc/P2U3-8BPU

[28] Although it is anticipated that change will be at a very low frequency

[29] Roy inspect on the Siegfried Wiki: https://perma.cc/PPM7-P6RA

3. *Container signature analysis*

There are some categories of signatures for which we need to create data models within Wikidata. Container signatures and multi-part signatures do not yet have the suitable models. The requirement was visited through the Siegfried/Wikidata collaboration. In-time we plan to propose additional combinations of properties and qualifiers to represent these types of signatures. As we do the entirety of the integration, we invite the digital community to share their own requirements on this front and consider what container and multi-part format identification might look like in a green-field development with multiple possibilities for extending both Wikidata, and Siegfried.

4. *Other experiments*

With Siegfried successfully connecting to the Wikidata Query Service it opens up possibilities to extract other information from Wikidata that extends what Siegfried is capable of reporting. One experiment was created to see how Siegfried might report on software compatible with a file format returned by an identification. An example output can be seen in table 2.

Siegfried at this point is using the benefits of linked open data to link the caller to other resources that help them engage with the content they are trying to preserve. A reference to a single piece of software might not be enough for most digital preservation workflows but it is easy to see from this example that the type of linking that can be done here could be out to a resource describing an entire software environment as originally envisioned in bwFLA[30].

At the time of writing this feature is just an experiment with more refinement needed if the concept is to arrive in Siegfried proper. It is hoped the demonstration is more important than the feature, inviting colleagues in the community to consider what information is now accessible through Wikidata and how that might be used or remixed to augment what utilities like Siegfried might return.

B. *Ecosystem/infrastructure*

This development project has laid the foundation for Wikidata to become more of a central part in digital preservation workflows. Around that an ecosystem needs to be developed which grows both out of the Digital Preservation and Wikidata communities.

Connecting Siegfried to Wikidata gives the digital preservation community an end-to-end workflow that enables the rapid input of signature information on an open platform and an immediate ability to then test the efficacy of the data input. In that regard, the integration represents something new that the community has not had so easily available in file format signature develop-

[30] A description from bwFLA for citing emulation environments using a handle system via the DPC website for the 2014 Digital Preservation Awards: https://perma.cc/6QQS-AN9X

```
---
filename : 'skeleton-test-suite/x-fmt-410-signature-id-197.exe'
filesize : 8
modified : 2020-07-08T23:53:57-04:00
errors   :
matches  :
  - ns       : 'wikidata'
    id       : 'Q4045294'
    format   : 'New Executable'
    URI      : 'http://www.wikidata.org/entity/Q4045294'
    mime     :
    basis    : 'byte match at [[0 2] [6 2]] (Wikidata reference is empty)'
    warning  : 'extension mismatch'
    software :
        Windows 8: 'http://www.wikidata.org/entity/Q5046'
        Windows 7: 'http://www.wikidata.org/entity/Q11215'
        Windows 98: 'http://www.wikidata.org/entity/Q483132'
        Windows 10: 'http://www.wikidata.org/entity/Q18168774'
```

Table 2: Potential software output in Siegfried using Wikidata

ment workflows before, with perhaps, the exception of the PRONOM signature development utilities[31][32] which can only be used for testing individual signatures in isolation, which is only a small part of the process.

For Wikidata to become a community governed registry for digital preservation then the digital preservation community needs to make use of the parts of it that will continue to improve the work being done in the GLAM sector. For file format identification then it will involve looking at the quality of the Wikidata output and then making incremental changes to what is recorded to make sure that it is of the utmost quality. It should involve defining what "quality" means, e.g. is avoiding file format signature conflicts - the ability for a file format to be matched against two different signature records considered as important for the community using the Wikidata registry as it is for the PRONOM governance. It should define how the Wikidata integration is expanded beyond the capabilities of a PRONOM one; e.g. what does container format identification look like in a Wikidata registry; is it good enough to transpose existing methods into Wikidata or are there ways to expand on what techniques are already available and push format identification into new territory. As discussed previously, Siegfried's abstract implementation lends itself to expanding on current capabilities.

Therefore, engagement is one of the key parts of making the integration a further success.

Additional tooling can be created in support of this effort as well. Where Siegfried provides some static analysis of the quality of data in Wikidata presently, the mechanism will need an overhaul once the module is put into greater use. Events increasing outreach or knowledge have also proven successful for other tooling in the past - as discussed, the inception of FFMprovisr or the development of JHOVE through its hack-days and weeks[33], likewise for PRONOM[34].

C. *Creating a "together" community*

The corollary to engagement with the integration is that the digital preservation community also needs to learn how to engage with Wikidata in ways that are sympathetic with the Wikidata community guides and guidelines. We are potentially two communities that need to come together, not a digital preservation community squatting in a Wikidata world.

Wikidata makes it very easy to edit entries. It offers a discussion platform to discuss those changes too. Where new properties are added to the Wikidata schema then those can be discussed in forums of increasing importance as well. While the service is community governed, that governance is somewhat different to the idea of a central point of focus to the registries of the past.

There is a certain amount of "Wikidata literacy" that needs to be developed alongside this resource to help make it work. In this sense two communities can become a "together" community with that of "digital preservation" branching out beyond our national memory institutions or vendors as leaders in our community registry work, to that of the wider technology field. An unintended consequence of that effort too is that the wider technology world may have resources to help bolster "our" efforts.

People who would like to contribute file format sig-

[31] Signature Development Utility 1.0: https://perma.cc/MH3L-EPM4

[32] Signature Development Utility 2.0 (ffdev.info): https://perma.cc/RDG9-N6BU

[33] Information about the OPF's (Open Preservation Foundation) JHOVE Hack Week 2019 via the DPC (Digital Preservation Coalition) website: https://perma.cc/W37P-DSSX

[34] Information about the PRONOM Hack Week 2020 via the OPF's website: https://perma.cc/H7S9-5LK4

nature information to Wikidata are welcome to add it. Digital Preservationists could share signature data via Wikidata, and could even use the platform to discuss with one another. Each item in Wikidata has an associated "Discussion" page" just as each article in Wikipedia does. If someone needs clarification about how a signature was determined it is possible to ask that question on Wikidata and see if others reply.

VIII. Conclusion

The risk of a file format registry is that it remains dormant - or that it contains a wealth of information that remains abstract and inaccessible. During the We Miss iPRES event of 2020 it was commented on that the Siegfried/Wikidata integration solved the issue of open registries and identification methods. That is true, but beyond that it also begins to make inroads into the idea of linked open data and file format registries. Here though, the data is not just linked to the web though HTTP URIs, it is linked to a user's file-system through Siegfried. When a Siegfried identification is returned through the Wikidata identifier it becomes a portal to the web that can then become a portal into your preservation plans and strategies - this data is actionable. It is not a filing cabinet of standards waiting to become stale. And that is also one of the reasons PRONOM has persisted where other registries have yet to do so[35].

It is a lesson that it is hoped will benefit the future of the concept of the Preservation Action Registry[36] - where key file formats, types of information, and digital record types can be computationally connected back to a resource, and therefore a workflow, that is not just the foundation of the sharing of information in a preservation action registry - but is also the key to automating those workflows - by connecting information not just through the web, but through organization's and individual's *file-systems*.

Acknowledgment

K.T. thanks Kenneth Seals-Nutt for his work to develop Wikidata for Digital Preservation. K.T. thanks Carl Wilson for hosting wikidp.org and his technical mentorship. K.T thanks the Su Lab of the Scripps Research Institute for creating WikidataIntegrator. K.T. thanks the entire Wikidata community for creating infrastructure and contributing data to the knowledge base. R.S. thanks to Kat and Euan and the Yale University Library team for the opportunity to work on this project. R.S. Thanks to Richard for creating Siegfried, his continued guidance, and being open to working together on this effort as well.

[35] As well as through the expert steerage and technical capabilities of the team at The National Archives, UK.

[36] Preservation Action Registry homepage: https://perma.cc/C9Z8-SXAX

References

[1] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 57, pp. 78–85, 2014. doi: 10.1145/2629489. [Online]. Available: https://web.archive.org/web/20190311200511/http://cacm.acm.org/magazines/2014/10/178785-wikidata/fulltext.

[2] A. Waagmeester, G. Stupp, S. Burgstaller-Muehlbacher, B. M. Good, M. Griffith, O. L. Griffith, K. Hanspers, H. Hermjakob, T. S. Hudson, K. Hybiske, *et al.*, "Science forum: Wikidata as a knowledge graph for the life sciences," *ELife*, vol. 9, e52614, 2020.

[3] Wikidata, *Statistics*, 2021. [Online]. Available: https://perma.cc/3CSH-DTYL.

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

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

[6] 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.

[7] K. Thornton, H. Solbrig, G. S. Stupp, J. E. L. Gayo, D. Mietchen, E. Prud'Hommeaux, and A. Waagmeester, "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.

[8] P. McKinney, D. Pearson, D. Anderson, J. Hutar, S. Knight, L. Coufal, J. Delve, J. Gattuso, K. DeVorsey, and R. Spencer, "A next generation technical registry: Moving practice forward," *iPRES 2014: 11th International Conference on Digital Preservation*, 2014.

[9] P. McKinney, S. Knight, J. Gattuso, D. Pearson, L. Coufal, D. Anderson, J. Delve, K. De Vorsey, R. Spencer, and J. Hutar, "Reimagining the format model,"

[10] J. Delve, *The Trustworthy Online Technical Environment Metadata database - TOTEM*. Hamburg: Kovac, 2012, isbn: 978-3-8300-6418-3.

[11] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, *et al.*, "The fair guiding principles for scientific data management and stewardship," *Scientific data*, vol. 3, p. 160 018, 2016.

[12] T. Pellissier Tanon and F. Suchanek, "Querying the edit history of wikidata," *Hitzler P. et al. (eds) The Semantic Web: ESWC 2019 Satellite Events. ESWC 2019. Lecture Notes in Computer Science*, vol. 11762, pp. 161–166, Oct. 2019. doi: 10.1007/978-3-030-32327-1_32. [Online]. Available: https://doi.org/10.1007/978-3-030-32327-1_32.