A PID is a promise
Versioning with persistent identifiers

Martin Matthiesen
CSC – IT Center for Science
Espoo, Finland
martin.matthiesen@csc.fi

Ute Dieckmann
University of Helsinki
Helsinki, Finland
ute.dieckmann@helsinki.fi

Abstract

We present the update process of a dataset using persistent identifiers (PIDs). The dataset is available in two different variants: for download and via an online web interface. During the update process, we had to fundamentally rethink as to how we wanted to use PIDs and version numbering. We will also reflect on how to effectively use PID assignment in case of minor changes in the large dataset. We discuss the roles of different types of PIDs, the role of metadata, and access locations.

1 Introduction

While other disciplines have been affected by reproducibility concerns as described in Baker (2016), this has so far not been the case in the Humanities. With the increasing use of statistical methods and automated data processing in the Digital Humanities and Computational Linguistics, this is likely to change and manifestos such as Munafò et al. (2017) will become more relevant.

Making data available in a persistent manner is one important aspect of making a dataset reusable for further research, but is also important for reproducibility of existing research. Publication principles such as FAIR (Wilkinson et al., 2016) emphasise the importance of persistent identifiers (PIDs) and descriptive metadata.

In an abstract sense, the role of PIDs is very clear: "Persistent identifiers allow different platforms to exchange information consistently and unambiguously and provide a reliable way to track citations and reuse." (Rueda et al., 2016, 40). In the same article, the authors warn: "Low-quality metadata, uncurated content, and a lack of internal and/or external organisation create repositories that are impossible to navigate or to obtain information from." (Ibid., 41).

Using PIDs consistently to avoid the aforementioned pitfalls turned out to be complex. In this paper, we explore in detail what using PIDs and descriptive metadata records means in practice when updating a large dataset.

The paper addresses the following areas in the design and construction of a CLARIN infrastructure:

- Recent tools and resources added to the CLARIN infrastructure
- Metadata and concept registries, cataloguing and browsing
- Persistent identifiers and citation mechanisms
- Web applications, web services, workflows
- Models for the sustainability of the infrastructure, including issues in curation, migration, financing and cooperation

Martin Matthiesen and Ute Dieckmann 2019. A PID is a Promise - Versioning with Persistent Identifiers. *Selected papers from the CLARIN Annual Conference 2018*. Linköping Electronic Conference Proceedings 159: 103–112.

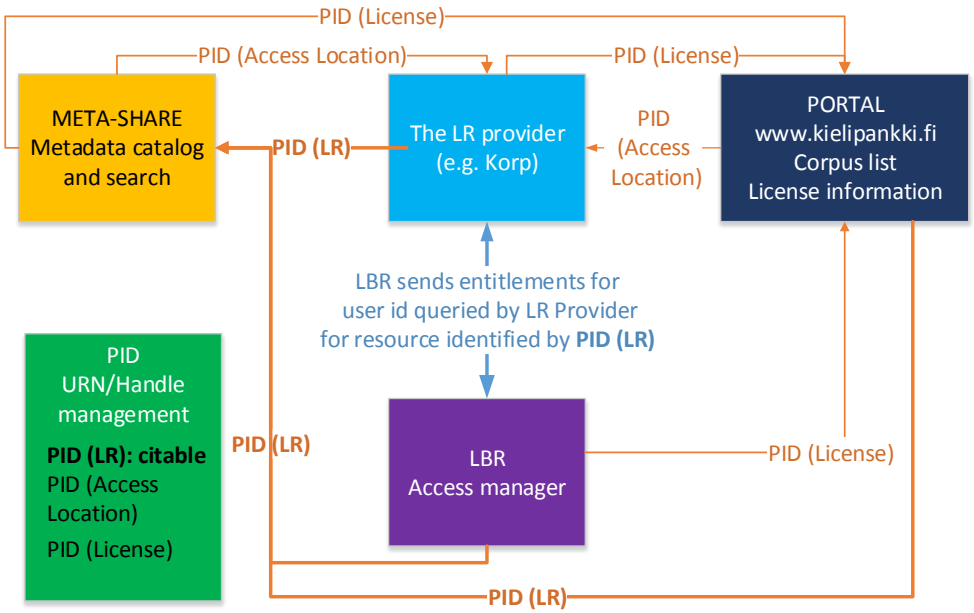


Figure 1: Resource and access management using PIDs

2 The repository

The Language Bank of Finland is a CLARIN B Centre and has therefore all the tools in place to provide data in a FAIR manner.

- A catalog for descriptive metadata using META-SHARE (http://metashare.csc.fi)
- A PID registry providing Handles and URNs.
- A download service (https://korp.csc.fi/download/)
- Corpus analysis tools such as Korp[1] (https://korp.csc.fi/)
- Access management via REMS[2] (Language Bank Rights, https://lbr.csc.fi/)
- Generated citation instructions of resources[3]

PIDs are used to reference language resources and implement access management. Figure 1 gives a general overview over the interaction of the components mentioned above.

In this paper, a resource consists of two main parts: the descriptive metadata in META-SHARE and the data itself. The role of the manually curated descriptive metadata (henceforth: metadata[4]) is to give the resarcher the "context information"[5] of the data itself.

We use the PID pointing to the META-SHARE metadata of a resource as the ID of the given resource. This PID (referenced as *PID(LR)* in figure 1) is citable and used in all services, such as Korp, Download, Language Bank Rights, and our corpus list to identify the resource. This citable PID is in fact the only essential PID needed to publish the dataset. The distinction between citable and non-citable PIDs is discussed further in section 7.

[1] Borin et al. (2012)
[2] See Linden et al. (2013) in Foster (2013)
[3] See "cite" column in our corpus list: https://www.kielipankki.fi/corpora/
[4] What we call a descriptive metadata page is sometimes referred to as "landing page" (See `https://documentation.library.ethz.ch/display/DOID/Landing+pages`.) We avoid the term in this paper.
[5] See Weigel et al. (2013).

At the Language Bank, PIDs are minted manually using a simple csv file as source in Github. Uniqueness is ensured by using the date of minting in reverse and a running number:

```
# Example
201801011 http://example-url.com
```

A script then registers URNs as well as Handles[6]. Handle attributes are not used for two reasons: compatibility with URNs, which do not support attributes, and the increased complexity of keeping the metadata in the attributes up-to-date and in sync.

3 The dataset

The dataset named "Corpus of Finnish Magazines and Newspapers from the 1990s and 2000s" was originally intended as an accruing dataset and therefore not versioned. The goal was to create a contemporary dataset of magazines and newspapers of various origins, such as scientific journals, regional newspapers, company internal circulations, and trade union member journals.

It was available for download licensed as CLARIN ACA +NC[7] and in Korp, licensed as CC BY[8].

We had preliminary policies in place for versioning accruing datasets[9]. In this case, we assumed that we would not need strict versioning for this dataset because changes seemed transparent enough for us as well as the research community. Over time it became apparent that using versioning is nevertheless more transparent than trying to avoid it.

3.1 Creation of the dataset

For this corpus, the original data was mostly harvested (partly automatically with the help of a python script) from the internet in PDF format. The PDF was converted to plain text with OCR software. These PDF and text files are available in our download service. For legal reasons, in a few cases we cannot provide the original PDFs.

The text files were then converted to the Corpus Workbench VRT format[10] using Python scripts. Structural attributes carrying metadata information, such as the name of the magazine, issue and date, were added. Finally the VRT data was enriched with dependency information, part-of-speech and named entity tags using the Turku Dependency Parser[11], and an earlier version of Finnish Tagtools[12]. This enriched VRT data was imported into Korp.

4 The initial update process

Even though the dataset consists of various individually identifiable newspapers and magazines, we had assigned only four PIDs to refer to the variants of the entire dataset: one PID to refer to the metadata of the Korp variant, one PID to refer to the metadata of the downloadable variant, and another two PIDs to point to the access location of the data itself, in Korp and our download service ("Download"), respectively.

Initially the dataset was updated as stated in the metadata and outlined in figure 2: frequently and without changing the PID. Information on the updates of each variant was maintained on a separate wiki page, referenced from the metadata. The metadata did not specify the update process of the variants. Korp and Download were not updated synchronously. Sometimes Korp would get updates before Download, more often it was the other way around.

[6] URNs and Handles can be derived from one another, this method developed at the Language Bank is now part of official GEDE/RDA recommendations, see assertion *PID-45* in Wittenburg et al. (2017, Section 3.3).

[7] `http://urn.fi/urn:nbn:fi:lb-2016050602`

[8] `https://creativecommons.org/licenses/by/4.0/`

[9] See The Language Bank's *Life cycle and metadata model of language resources*: `http://urn.fi/urn:nbn:fi:lb-201710212`

[10] See `https://www.kielipankki.fi/development/korp/corpus-input-format/` for a more detailed description.

[11] `http://turkunlp.github.io/Finnish-dep-parser/`

[12] See University of Helsinki (2018).

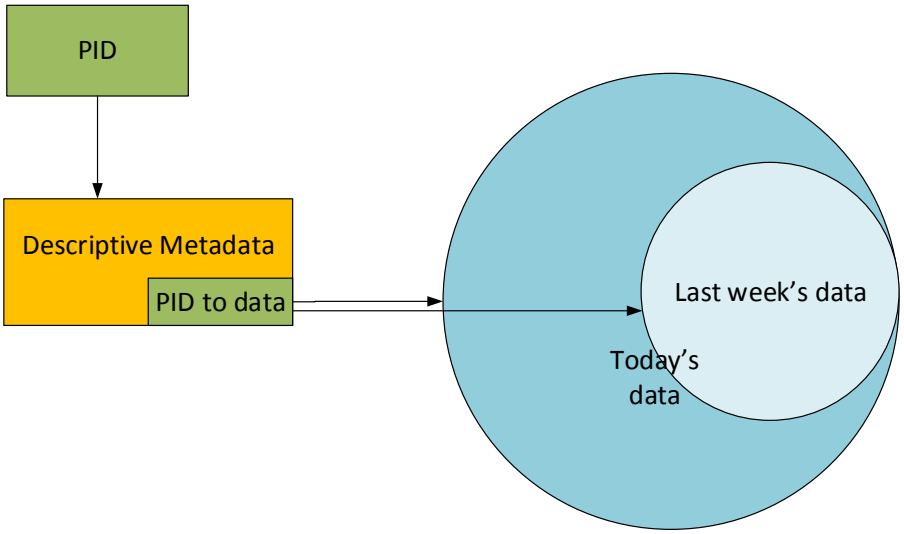
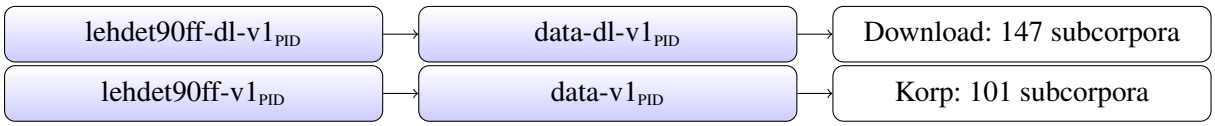


Figure 2: Corpus variant with an unversioned PID

The PID pointing to the top level directory of the Download variant of the dataset would automatically include any added content. This was not the case with Korp. The PID pointing to the Korp collection was not consistently updated, explicitly selecting only parts of the dataset for use in Korp.

However, even the extended versions of the dataset were implicitly addressed, since new subcorpora showed up as additional selectable items under the same collection in Korp. At the time of the update, we had a corpus of 147 subcorpora in Download and 101 subcorpora in Korp. Figure 3 shows how the citable PID points to the metadata and the metadata in turn points to the access location of the actual data.



lehdet...$_{PID}$: A simplified persistent identifier pointing to the descriptive metadata
data...$_{PID}$: A simplified persistent identifier pointing to the data itself
Korp,Download: The access location of the data itself

Figure 3: PIDs and access locations for version 1 before the update

During the update, we discovered issues in both dataset variants:

- Some subcorpora in Korp and Download were missing data, due to previously unnoticed problems with the conversion.
- Some Korp subcorpora were not properly annotated.
- Some Download zip files did not have license and README information.
- Existing README/license.txt files were located in the root path of zip files, and they were overwritten if more than one zip file was unzipped in the same directory.
- The directory structure of the zip files was generally not consistent.
- Files zipped on a Mac had filename encoding problems in Linux.

- Some zip files contained thumbnails and other irrelevant temporary files/directories.

In other words, an update planned as a simple addition of data turned into the curation of an already published dataset.

5 A more consistent approach

At the time of the update, the dataset was by design unversioned. The variants in Korp and Download were not synchronized, and existing data in both variants needed to be curated. We essentially faced a versioning task, as described in appendix A3 in Weigel et al. (2015, 21). The decisions we made are explained below.

5.1 Versioning

First, we abandoned the idea of an accruing dataset behind a single PID. It is clear what the PID denotes at any given point in time, and its general intension stays the same. Determining the concrete extension of such a PID at different times is possible, but impractical and error prone. We therefore now take the temporal component into account and introduce versioning as shown in figure 4.

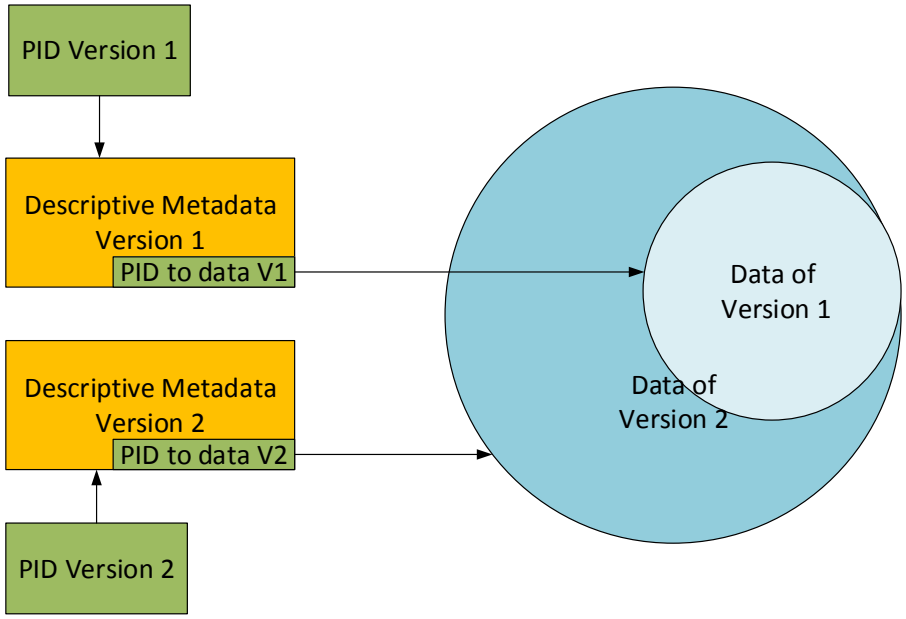


Figure 4: Corpus variant with a versioned PID

Versioning for small datasets in textual format is largely solved if they do not significantly exceed the size of a computer program[13]. Version control for computer software like Github[14] has been around for a long time. In our case, we could not use it for many reasons, the most important one being the size of the dataset. We had to invent our own approach.

As the two variants (Korp and Download) of the dataset were not in sync, we briefly considered synchronizing them to have a well-defined starting point for the update. We abandoned this idea, since it would require even more PIDs and version numbers. So we accepted that we did not have a well defined starting point and aimed for a well-defined end state.

[13] To our knowledge only the *Hamburger Zentrum für Sprachkorpora* uses *git* for versioning of text corpora, see `https://inel.corpora.uni-hamburg.de/wp-content/uploads/jettka_hedeland-2018-HZSK_INEL_Workflows.pdf`

[14] `https://github.com`

We therefore introduced version 1 of each variant and made it explicit that they are overlapping but not absolutely in sync. The updated and synchronized dataset, version 2, now contains 369 subcorpora in either of its two variants.

5.2 Stop-over pages

We use PIDs for metadata resolution and resource resolution, as defined in Weigel et al. (2013). In our case, the Korp variant of version 1 was not worth keeping online unchanged. For example, some subcorpora lacked part-of-speech information in version 1 that was added in version 2, but the content was otherwise unchanged. A search performed on version 1 can thus be repeated with version 2 by simply ignoring the part-of-speech information. In other cases, attributes were renamed. Again, the old search could be repeated by slightly modifying it to work with version 2. Since we had quite a few such changes, we decided not to keep version 1 online, and instead point the resource PID of version 1 to the relevant subset of version 2. To make the changes transparent, we did not point the PID directly to the resource, but to a "stop-over page".

A "stop over page" is a manually curated web page accessed by a resource PID that has pointed to data which is not available in its original form any longer. The changes are explained and the user is directed further to the location of the corrected data, as outlined in figure 5. The stop-over page either gives access to the previously available data or it provides information on how to use the updated data to get comparable results. A stop-over page shares properties with a tombstone page. Both refer to data not directly available anymore. A tombstone page is used when it is hard or impossible to recreate the old data.

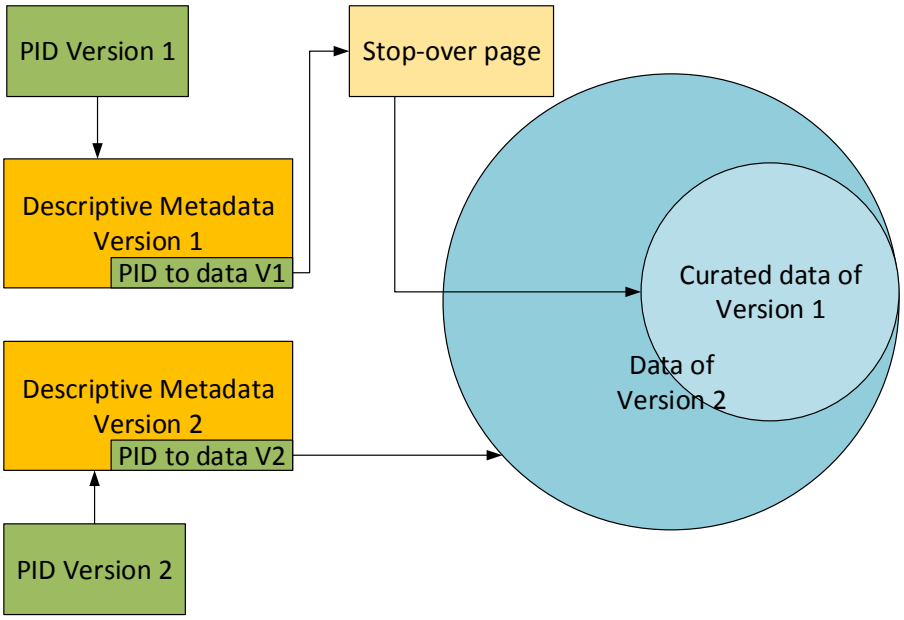


Figure 5: Corpus variant with a versioned and curated PID

The stop-over page makes it possible to take old data offline without excessively compromising reproducibility. This not only saves storage and backup space, but also improves accessibility and structure of the repository as a whole.

5.3 Change Log

The Change Log is a well known concept in software engineering[15]. We use it to describe "minor changes" in the dataset that we do not consider worth a version change. For example[16]:

```
Change Log:
20.2.2018 Name and file type of "tiedelehdet_terminfo.vrt.gz" changed to
          "Terminfo 2010-2015.zip" (subdirectory added in zip file)
```

The Change Log is kept in the metadata record of the dataset and only relates to the version at hand, unlike software changelogs that often contain the whole version history. In our case, that history is kept using relations between metadata records. For those relations (e.g. *IsPreviousVersionOf*), we use a subset of the controlled vocabulary described in DataCite Metadata Working Group (2016).[17]

5.4 PID granularity

During the update we also considered changing the granularity of the PIDs. In the following sections, we explain why we did not opt for increasing the amount of PIDs.

5.4.1 Rejecting data object PIDs

We evaluated the introduction of data object PIDs. The CLARIN B Centre Requirements state that data objects can be assigned a PID if they "are considered to be worth to be accessed directly (not via metadata records) by the data provider" (Wittenburg et al., 2018, Section 7).

Such data object PIDs are obviously useful for machine to machine communication, they can be accessed by scripts and automatic processing pipelines. However, this is of practical use only for small datasets. It is fair to assume that large datasets will hardly ever be processed online, but rather downloaded, decompressed and processed locally. Data object PIDs are of little use in such a scenario.

In our data curation task we had to make at least minor changes to all subcorpora in Korp and Download. Version 1 of the downloadable corpus (University of Helsinki, 2017) alone is a collection of 147 subcorpora consisting of 413 zip files and tens of thousands of individual files.

Had we assigned 413 PIDs to the zip files, most of them would have needed stop-over pages, because we changed the content of the zip files by adding READMEs and subdirectories, correcting typos in filenames, and so on. It would not have been feasible to keep the old zip files online. Any script relying on the PIDs would have stopped working at this point. Even if the stop-over page had been machine readable, the end result would have been that the old zip file would not have been provided automatically.

PIDs to individual files would have required us to provide the content either uncompressed or compress the files individually and would have created a need for even more stop-over pages. Storage and bandwidth considerations also had to be taken into account. Apart from the higher maintenance need for hundreds of PIDs, we did not see an added value for a user using only a subset of the corpus. The subset can still be defined relative to the dataset variant referenced by the PID.

Instead, we use one PID for the Download variant and explain the changes in the metadata in a Change Log. We also maintain a Change Log in the metadata and a stop-over page to explain the changes we made to the already published subcorpora in Korp. We used the stop-over page only for the Korp variant, since we considered the changes in Download minor enough to be described in the Change Log.

5.4.2 Adequate PID granularity

In the previous chapter, we argued that as few PIDs as possible should be used. As shown in figure 6, we are down to four, two per dataset variant, two for each version. Why not even less, why not use one PID for both variants, reducing the number of PIDs further to two, one for version 1 and one for version 2?

The Prague Dependency Treebank is published in this way: The PID points to the metadata from where the data can be dowloaded or accessed with two distinct web based tools[18]. We considered two use cases

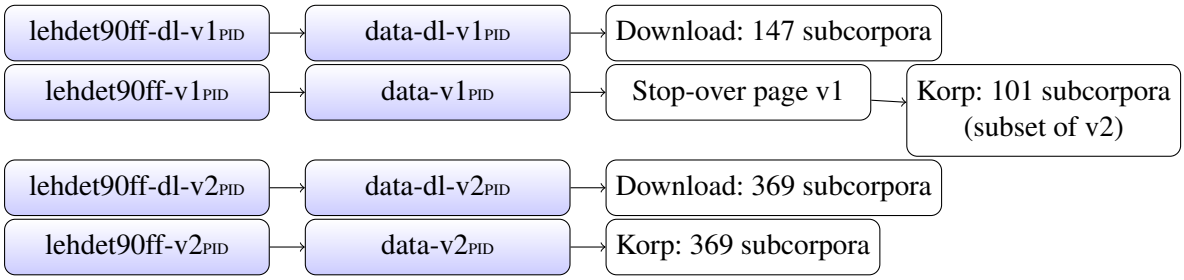
[15] https://www.gnu.org/prep/standards/standards.html#Change-Logs

[16] See `http://urn.fi/urn:nbn:fi:lb-2017091902`

[17] See The Language Bank's *Life cycle and metadata model of language resources*: `http://urn.fi/urn:nbn:fi:lb-201710212`

[18] Cf. (Hajič et al., 2018).

and decided that both warrant separate PIDs: If a user uses only one instance of a corpus, for example only the downloadable version or only the Korp web interface, then there will be only one reference in any case. Should a user use both variants in one paper, different PIDs make it easier to keep track of what has been observed and where. After all, it is always possible that the variants differ in unintended ways.



lehdet...PID: A simplified persistent identifier pointing to the descriptive metadata
data...PID: A simplified persistent identifier pointing to the data itself
stop-over page: As described in section 5.2
Korp,Download: The access location of the data itself

Figure 6: PIDs and access locations after the update

6 Generated PIDs vs. manual PID curation

Automatic management of PIDs has a few advantages: Links and relations are created in a consistent manner. Data changes can be detected automatically and new PIDs can be instantly created, if needed. This is especially important when dealing with a large number of PIDs.

However, the automatic approach cannot easily distinguish between significant and non-significant changes, as for example adding a missing comma in the README.txt file of a downloadable dataset. Keeping the old dataset in this case and minting a new PID makes no sense, at least not in terms of reproducibility or responsible usage of storage space. Changing a character in the tagset of a corpus can be a minor or major change. The computer cannot yet categorize changes and more importantly cannot substantiate and justify such categorizations; this still needs to be done by humans.

The tools we use, such as META-SHARE do not support automatic PID handling. Minting them manually as described in section 2 gives us more flexibility in using them. It does, however, also leave room for inconsistencies, as discussed with inconsistent updates of data PIDs in section 4.

While fully automatic PID handling is not desirable, automatic checking of existing PIDs and their relations would help to ensure more consistency. The usability of META-SHARE would also benefit from better support for PIDs and expressing their relations using controlled vocabularies, as suggested by DataCite Metadata Working Group (2016).

7 Citable vs. non-citable PIDs

We divide PIDs into two major categories, regardless of the underlying PID resolver technology (eg. URN, Handle, DOI): Citable and non-citable. Citable PIDs point to the authoritative metadata of the resource, and therefore are absolutely essential properties of a dataset. It cannot be published without them being assigned. Non-citable PIDs can be used to refer to the access location of the data itself. Non-citable PIDs are not absolutely necessary since the dataset can always be referenced using its citable PID. They can be further subdivided into access location and data object PIDs. While not essential, they can be useful to manage changes in access locations, such as server name changes.

- Citable PID: PID to authoritative metadata of a resource
- Non-citable PID
  - Access location PID (points to the actual data location in services such as Korp or Download)

- Data object PID (directly points to data object, like zip, pdf, wav, mp4)
- PIDs to license pages

Note that stop-over pages are also useful in scenarios where only citable PIDs are used. In that case, the direct access location link to a dataset is replaced by a link to a stop-over page leading further to the updated dataset access location.

8 Discussion and Conclusions

Our aim was to update two variants of a previously unversioned dataset in a way that enables researchers to replicate earlier studies. Transparent information should be provided on any deviations within each version.

We created our own approach to versioning. In section 5.4.1, we showed that it is often not practical to keep earlier versions of large datasets available online. Taking a dataset offline immediately breaks automatic workflows. It would also break data object PIDs, which is one reason why we consider them impractical.

We showed that the inflationary automatic creation of PIDs (usually to data objects) considerably increases curation needs. The consequences in terms of human and technical resources can be significant. By making a clear distinction between mandatory citable and optional non-citable PIDs, we offer a way to keep the focus in PID handling.

A PID is, not unlike a bank note, a promise. Once you create it, you have to make sure it keeps its value.

Also not unlike currency, different people see different values in PIDs. In our opinion, the core value of a PID is the ability to make datasets traceable, even if they change over time and older versions are not available online anymore.

Our aim is not to ensure 100% repeatable runs of scientific software over the span of many years. Our aim is to enable plausible repeatability of research. Such repeats might require changes in the original scientific code or web request to produce similar outputs. To make possible changes transparent we introduced a Change Log and the new concept of stop-over pages.

To sum up, when introducing versioning using PIDs, we tried to find a balance between maintainability, usability, and transparency at every stage of the update process of a large dataset.

9 Outlook

While we do not want to maintain data object PIDs, a direct path to a data object via a general PID is useful. The transparent implementation of part identifiers for URNs and Handles[19] would be a solution to this problem. Automatic validation of manually created PIDs and their relations is another area of improvement. META-SHARE could work more with controlled vocabularies and warn of missing back-references in case of reciprocal relations. The efficient storage, versioning, and dissemination of large binary datasets continues to be a challenge. We intend to evaluate efforts like the German KA3 Project[20] for its applicability to our data management needs.

References

Monya Baker. 2016. 1,500 scientists lift the lid on reproducibility. *Nature*. https://doi.org/10.1038/533452a.

Lars Borin, Markus Forsberg, and Johan Roxendal. 2012. Korp the corpus infrastructure of Språkbanken. In *Proceedings of LREC 2012. Istanbul: ELRA*, page 474478.

DataCite Metadata Working Group. 2016. Datacite metadata schema for the publication and citation of research data v4.0. page 37ff. https://doi.org/10.5438/0012.

[19] See assertion *PID-45* in Wittenburg et al. (2017, section 3.3)

[20] http://dch.phil-fak.uni-koeln.de/ka3.html, in German

David Foster, editor. 2013. *Innovating Together, The 29th Trans European Research and Education Networking Conference, 3 - 6 June, 2013, Maastricht, Netherlands, Selected Papers*. TERENA, August. http://www.terena.org/publications/tnc2013-proceedings/.

Jan Hajič, Eduard Bejček, Alevtina Bémová, Eva Buráňová, Eva Hajičová, Jiří Havelka, Petr Homola, Jiří Kárník, Václava Kettnerová, Natalia Klyueva, Veronika Kolářová, Lucie Kučová, Markéta Lopatková, Marie Mikulová, Jiří Mírovský, Anna Nedoluzhko, Petr Pajas, Jarmila Panevová, Lucie Poláková, Magdaléna Rysová, Petr Sgall, Johanka Spoustová, Pavel Straňák, Pavlína Synková, Magda Ševčíková, Jan Štěpánek, Zdeňka Urešová, Barbora Vidová Hladká, Daniel Zeman, Šárka Zikánová, and Zdeněk Žabokrtský. 2018. Prague dependency treebank 3.5. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University. http://hdl.handle.net/11234/1-2621.

Mikael Linden, Tommi Nyrönen, and Ilkka Lappalainen. 2013. Resource Entitlement Management System. In Foster (Foster, 2013). http://www.terena.org/publications/tnc2013-proceedings/.

Marcus R. Munafò, Brian A. Nosek, Dorothy V. M. Bishop, Katherine S. Button, Christopher D. Chambers, Nathalie Percie du Sert, Uri Simonsohn, Eric-Jan Wagenmakers, Jennifer J. Ware, and John P. A. Ioannidis. 2017. A manifesto for reproducible science. *Nature Human Behaviour*, 1:21, Jan. http://dx.doi.org/10.1038/s41562-016-0021.

Laura Rueda, Martin Fenner, and Patricia Cruse. 2016. Datacite: Lessons learned on persistent identifiers for research data. *International Journal of Digital Curation*, 11(2). https://doi.org/10.2218/ijdc.v11i2.421.

University of Helsinki. 2017. Corpus of Finnish Magazines and Newspapers from the 1990s and 2000s, Downloadable Version 1. The Language Bank of Finland. Retrieved from http://urn.fi/urn:nbn:fi:lb-2016050401.

University of Helsinki. 2018. Finnish Tagtools. The Language Bank of Finland. Retrieved from http://urn.fi/urn:nbn:fi:lb-2018062101.

Tobias Weigel, Michael Lautenschlager, Frank Toussaint, and Stephan Kindermann. 2013. A framework for extended persistent identification of scientific assets. *Data Science Journal*, 12:10 – 22. https://doi.org/10.2481/dsj.12-036.

Tobias Weigel, Timothy DiLauro, and Thomas Zastrow. 2015. PID Information Types WG final deliverable. Technical report, Research Data Initiative. https://doi.org/10.15497/FDAA09D5-5ED0-403D-B97A-2675E1EBE786.

Mark D. Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Tim Clark, Mercè Crosas, Ingrid Dillo, Olivier Dumon, Scott Edmunds, Chris T. Evelo, Richard Finkers, Alejandra Gonzalez-Beltran, Alasdair J. G. Gray, Paul Groth, Carole Goble, Jeffrey S. Grethe, Jaap Heringa, Peter A. C. 't Hoen, Rob Hooft, Tobias Kuhn, Ruben Kok, Joost Kok, Scott J. Lusher, Maryann E. Martone, Albert Mons, Abel L. Packer, Bengt Persson, Philippe Rocca-Serra, Marco Roos, Rene van Schaik, Susanna-Assunta Sansone, Erik Schultes, Thierry Sengstag, Ted Slater, George Strawn, Morris A. Swertz, Mark Thompson, Johan van der Lei, Erik van Mulligen, Jan Velterop, Andra Waagmeester, Peter Wittenburg, Katherine Wolstencroft, Jun Zhao, and Barend Mons. 2016. The FAIR guiding principles for scientific data management and stewardship. *Scientific Data*, 3:160018, Mar. http://dx.doi.org/10.1038/sdata.2016.18.

Peter Wittenburg, Margareta Hellström, Carlo-Maria Zwölf, Hossein Abroshan, Ari Asmi, Giuseppe Di Bernardo, Danielle Couvreur, Tamas Gaizer, Petr Holub, Rob Hooft, Ingemar Häggström, Manfred Kohler, Dimitris Koureas, Wolfgang Kuchinke, Luciano Milanesi, Joseph Padfield, Antonio Rosato, Christine Staiger, Dieter van Uytvanck, and Tobias Weigel. 2017. Persistent identifiers: Consolidated assertions. Status of November, 2017., December. https://doi.org/10.5281/zenodo.1116189.

Peter Wittenburg, Dieter Van Uytvanck, Thomas Zastrow, Pavel Strak, Daan Broeder, Florian Schiel, Volker Boehlke, Uwe Reichel, and Lene Offersgaard. 2018. CLARIN B Centre Checklist. Technical Report CE-2013-0095, CLARIN ERIC. http://hdl.handle.net/11372/DOC-78.