

Recommendations on Data Versioning

Authors: Jens Klump^{1,2}, Heinz Pampel², Laura Rothfritz², Dorothea Strecker²

¹Mineral Resources, CSIRO, Perth WA, Australia

²Berlin School of Library and Information Science, Humboldt-Universität zu Berlin, Berlin, Germany

Suggested citation:

Klump, J., Pampel, H., Rothfritz, L., & Strecker, D. (2024). Recommendations on Data Versioning (p. 26). Berlin, Germany: Berlin School of Library and Information Science, Humboldt-Universität zu Berlin. <https://doi.org/10.5281/zenodo.13743876>

Table of contents

Introduction	4
Revisions	5
File-based revisions management - Systematic names	5
File-based revisions management	6
Database timestamps and revisions	6
Recommendations	7
Releases	7
Significance vs. volume of change	8
Recommendations	9
Granularity (Items)	9
Collections	9
Time series	11
Recommendations	11
Mirroring and re-publication	11
Recommendations	12
Deleted items and Retractions	12
Item deleted	12
Metadata deleted	12
Data publication retracted	12
Recommendations	13
Provenance	13
Recommendations	13
Formats (Manifestations)	13
Recommendations	14
Citation	14
Versioning of Metadata	14
Recommendations for metadata versioning	15
Choosing a Versioning System	15
Questions to Ask	16
Related Concepts	17
Functional Requirements for Bibliographic Records	17
Data Life Cycle and Domains of Responsibility	18
Private Domain	18
Collaboration Domain	19
Publication Domain	19
Open Archival Information Systems	19
Designated User Community	20
References	20

Background

This recommendation was developed as part of the "PID Reference Model for Versioning Research Data" (PIDsBUA) project at the research and teaching group Information Management at the Berlin School of Library and Information Science, located at Humboldt-Universität zu Berlin.

The project deals with the identification and versioning of research data, which are made accessible and reusable in digital repositories, and its impact on referencing and citation in accordance with the FAIR principles, with special consideration of Open Science practices.

This recommendation outlines key aspects of research data versioning for scientists and information management professionals at research-performing organisations.

The recommendations are based on previous work by the Research Data Alliance Data Versioning Working Group.

The project was funded in 2024 under Objective 3, "Advancing Research Quality and Value," of the Berlin University Alliance (BUA).

Introduction

We often say that “A is a version of B” but do not explain what we mean by “version”. We imply that B was somehow derived from A or that they share a common ancestor. But how is B related to A? How do they differ? Do they differ in content or format? What is the significance of this difference? While this sounds like a question about the provenance of a dataset, it goes beyond that and asks questions about the identity of a digital object and the intellectual and creative work it embodies.

Versioning practices have been standard in software development for decades (Software versioning, 2019), and new practices have been developed. The Research Data Alliance Data Versioning Working Group (<https://www.rd-alliance.org/groups/data-versioning-ig/>) collected over forty use cases of versioning practices for data and software (Klump et al., 2020a) and published a set of principles distilled from the group's analysis of the use cases (Klump et al., 2021, 2020b). For readability, this set of principles will be referred to as the “Principles” in this document. The Principles define terminology that helps us differentiate different types of versioning and thus allow us to address the use cases more precisely.

In follow-up discussions, we learned that the Principles are too abstract to apply to the operation of data repositories or to guide the citation of digital resources. Therefore, this document aims to translate the Principles into actionable recommendations for data versioning.

The Principles are built upon prior work in other data and information management areas. In particular, the Principles draw upon concepts developed in the Functional Requirements for Bibliographic Records (FRBR) (O'Neill, 2002), the Open Archival Information Systems Reference Model (OAIS) (CCSDS, 2012), and an analysis of current practices for the use of Persistent Identifiers (PID) (e.g. Klump et al., 2016).

General Recommendations

Adopt a Consistent Versioning Strategy: Implement a standardised approach to versioning research data across your organisation to ensure consistency and clarity. This should include clear guidelines on when and how new versions are created and documented.

Consider standardisation initiatives: Consider ongoing standardisation initiatives and incorporate established and emerging practices, as well as frameworks like those developed by the Research Data Alliance (RDA). This ensures consistency, clarity, and alignment with best practices in the research community.

Use Persistent Identifiers (PIDs) for Versioning: Assign persistent identifiers (PIDs) to each version of your datasets. This ensures that every version is uniquely identifiable and accessible over time, improving data traceability and citation.

Implement Clear and Descriptive Version Labels: Use clear, descriptive, and consistent labels for different versions of datasets. This includes adopting widely recognised conventions for data releases and maintaining transparency in interim data revisions.

Ensure User-Friendly Version Control Systems: Design or select version control systems that are intuitive and easy to use for all researchers, regardless of their technical expertise. The system should minimise the need for users to understand its underlying mechanisms.

Document Changes and Metadata: Maintain thorough documentation of changes between dataset versions, including detailed metadata. This information should be accessible to all users to facilitate understanding of what has changed between versions.

Communicate Versioning Practices to Stakeholders: Regularly inform all stakeholders, including researchers, data managers, and collaborators, about the versioning practices and guidelines. This ensures everyone is aware of how to access and reference the correct versions of datasets.

Review and Update Versioning Practices: Regularly review and update your versioning practices to keep pace with technological advancements and changing organisational needs. Engage with the community to ensure that your practices align with emerging standards.

Revisions

The most basic case of versioning digital objects is noting a change in the bitstream of an object, which is called a **revision**. These changes can easily be detected technically, and several technical and non-technical solutions exist to deal with them. Version information makes a revision of a dataset uniquely identifiable, allowing data users to determine whether and how data has changed over time and to determine specifically which version of a dataset they are working with.

Version control systems should be designed so that the designated users are not required to understand your version control system. To make sense of the differences between multiple versions, you should explicitly mark official "releases" for your data using well-adopted and widely understood conventions, even if versions between the releases are tagged using a custom scheme (Buttigieg et al., 2022).

File-based revisions management - Systematic names

The simplest implementation approach is to give files **systematic names**. Data Carpentries (Martinez 2017) recommends this approach as the technically simplest. This method is suitable for individual researchers or small working groups.

To illustrate the pattern, Data Carpentries gives the following example:

```
2013-10-14_manuscriptFish.doc
2013-10-30_manuscriptFish.doc
2013-11-05_manuscriptFish_intititalRyanEdits.doc
2013-11-10_manuscriptFish.doc
2013-11-11_manuscriptFish.doc
2013-11-15_manuscriptFish.doc
2013-11-30_manuscriptFish.doc
2013-12-01_manuscriptFish.doc
2013-12-02_manuscriptFish_PNASsubmitted.doc
2014-01-03_manuscriptFish_PLOSsubmitted.doc
2014-02-15_manuscriptFish_PLOSrevision.doc
2014-03-14_manuscriptFish_PLOSpublished.doc
```

The naming convention uses the date of the file's modification in ISO format, which allows for simple file sorting by date and simplifies file searches.

When dealing with several sets of files, e.g. for different projects or experiments, the pattern can be extended to the folder in a file system.

File-based revisions management - Technical systems

An alternative to managing files in a file system is to use a versioning system based on Git (Martinez, 2017). Data Carpentries recommends this method because it supports versioning, backup, and file sharing.

Git has limitations on the file size it can handle. Data Version Control (DVC, <https://dvc.org/>) is designed to handle large files, data sets, machine learning models, metrics, and code. DVC allows for the storage and versioning of source data files, machine learning models, and intermediate results with Git without checking the file contents in Git. It is useful when dealing with files that are too large for Git to handle.

Database timestamps and revisions

The Working Group on Data Citation (WGDC) of the Research Data Alliance (RDA) developed and published recommendations on the citation of dynamic data. The aim is to provide a tool for describing which subset of a large dynamic dataset was used at a particular time. The group published a compact 2-page flyer (Rauber et al., 2015) and a peer-reviewed paper with more detail (Rauber et al., 2021).

The recommendations on the citation of dynamic data (Rauber et al., 2021) can be grouped into four areas:

- Preparing the Data and the Query Store
- Persistently Identifying Specific Data Sets
- Resolving PIDs and Retrieving the Data
- Modifications to the Data Infrastructure

A central component of this set of recommendations is that access to data is based on a query sent to a data access system. The data queries addressing a particular subset are stored and can be executed again in the future.

This approach is suited for systems like relational databases and versioning of file-based data via Git, XML databases, and NoSQL databases.

Recommendations

- File-based data (file system): use systematic file and folder names. Include the modification date of the file in ISO format. This approach is suitable for the private and collaboration domains.

- File-based data (git): use git for small files or DVC for larger files. This approach is suitable for the collaboration domain as it supports data sharing, documentation, and recording of provenance.
- Dynamic data (relational databases): follow the recommendations of the RDA Dynamic Data WG.

Releases

Datasets may undergo several changes before they are made available to other users. This editorial process of making a dataset available to other users is called a **release**. The dataset should be accompanied by a release note explaining the changes and their significance to a designated user community.

There are many reasons for making new releases available. Examples are:

- Fixing of errors that have been reported by users of the data
- A revision of the whole data set where a new approach has been applied to data cleaning of the original data
- A revision of the whole data set where a new approach has been applied to data processing (a new algorithm, for example) of the original data
- Data should be removed because certain properties have been identified as holding sensitive or confidential data (after release).
- Existential angst resulted in the author's decision to change the vision of their work significantly.

The significance of the changes should be published as a data release and explained in a release note using well-adopted and widely understood conventions, even if versions between the releases are tagged using a custom scheme (Buttigieg et al., 2022).

Figure 1 illustrates a case where an original data set (time = T_0) was appended and partially overwritten with changes. The proposed procedure is to publish a new data release after the first set of changes (time = T_1) and the second set of changes (time = T_2). Each release should be separately identified. In addition, to enhance findability, all releases can be identified by an umbrella object that encompasses all releases.

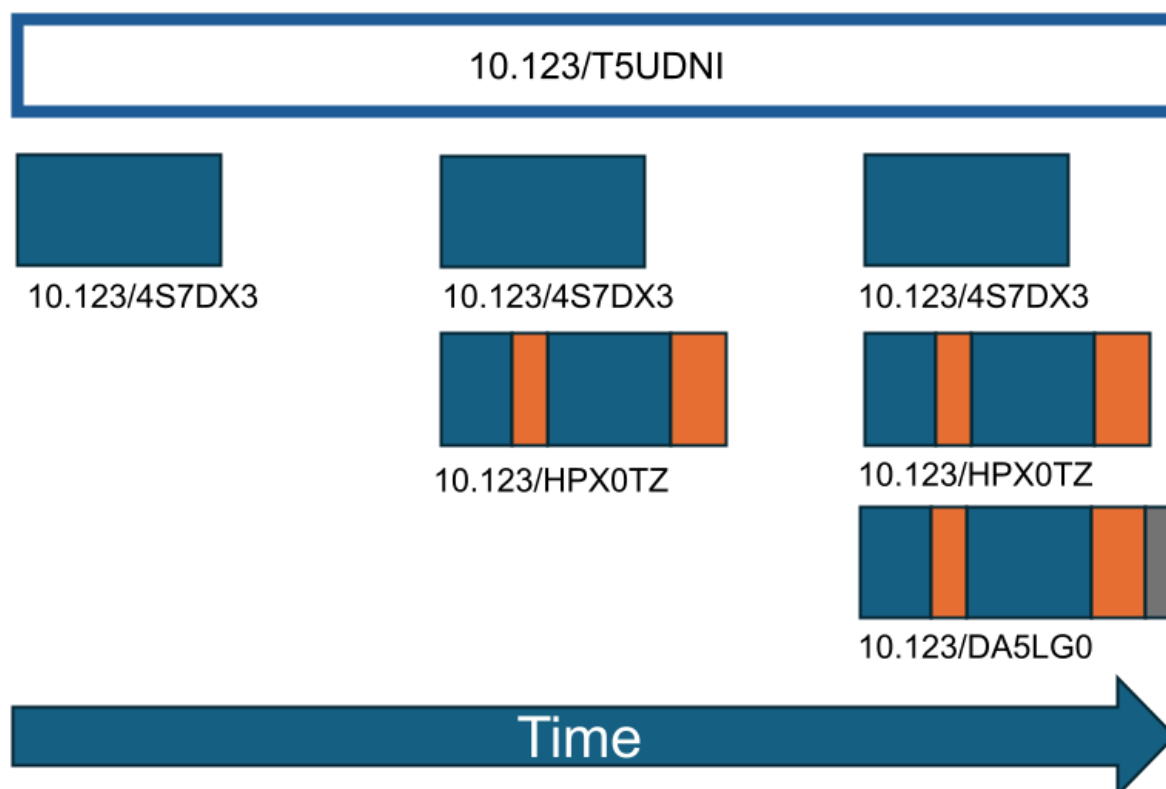


Figure 1: Identification and retention of data releases (Klump, 2024).

Significance vs. volume of change

While the identification and management of revisions is a primarily technical process that can be automated, the identification of revisions for release is an editorial and curatorial process in which the editor or curator decides that a revision is identified as a new **release**. The release should be accompanied by **release notes** (Lacan, 2023).

The significance of a change is not directly related to the magnitude of the change. Sometimes, small changes can be insignificant or highly significant. The significance also depends on the designated community.

Example: as a single datum, the change of a value from 3.14 to 31.4 as a string of characters has a Hamming Distance (Hamming, 1950) of $d=2$ but a mathematical change by one order of magnitude. On the other hand, in a set of one hundred random numbers between 0 and 99.9, the change would shift the mean value of all numbers in this set by about 0.5%, which might be insignificant. The significance of this change will depend on the use case.

Example: in AD 325, the First Council of Nicaea assembled to discuss the nature of Christ, whether Christ was the same as God (ὁμοούσιος) or similar to God (ὁμοιούσιος). Again, the Hamming Distance between the two strings is $d=2$, with both terms differing only by one iota. Nevertheless, this difference led to the first

schism of the Christian church. Since then, the designated community has changed; today, this question is only relevant to a small minority of Christian denominations.

Recommendations

- Releases: Establish workflows for creating and identifying data releases.
- Release notes: Use release notes to communicate the significance of the change and record the provenance of data.

Granularity (Items)

A data product (expression) can be a set of data items, and the number of files (items) associated with this data product can be very large. A data product might also change over time as a time series. Also, the data might come in different formats (manifestations) and can be made available at multiple locations (items). While it is important to make those differentiations, standards or best practices for references do not exist. The section below discusses different forms of data granularity and recommends how to refer to the respective instances.

The granularity, the smallest identifiable unit by a persistent identifier, is not always easy to define. In some cases, where time or space are dimensions of the data, a subset may be defined by a range or by bounding parameters, or data may have an internal structure that allows the formulation of a canonical path to the defined granule. In other cases, the data product may be composed into a collection of discrete objects (items).

Collections

A data product (**expression**) can be a set of data items, and the number of files (**items**) associated with this data product can be very large. In the context of data versioning, a collection is a set of discrete objects that together form a data product (Diepenbroek et al., 2014; Jenkyns et al., 2024).

Access to research data plays a key role in reproducibility. To enable reuse, a dataset's granularity must be comprehensible for its designated user community in the context of other outputs like the research publication, code, and protocol (Jenkyns et al., 2024). In particular, where studies draw on parts of existing datasets, the ability to reproduce the data depends on consistent granularity.

Since a collection is a set of discrete objects, each item could be identified individually. Some data centres do not apply individual PIDs to all items but use canonical paths or mint PIDs on demand. Some data centres use DOI as PIDs at the collection level (Klump et al., 2016) and alternate PIDs, e.g. EPIC IDs (Kálmán et al., 2012), at the individual item level. The use of alternate PIDs as part of data releases bears the risk of creating confusion about the identity of an object, especially if there is no way for machines to disambiguate the objects.

Figure 2 illustrates how a collection of objects can be identified, with each object having its own identifier.

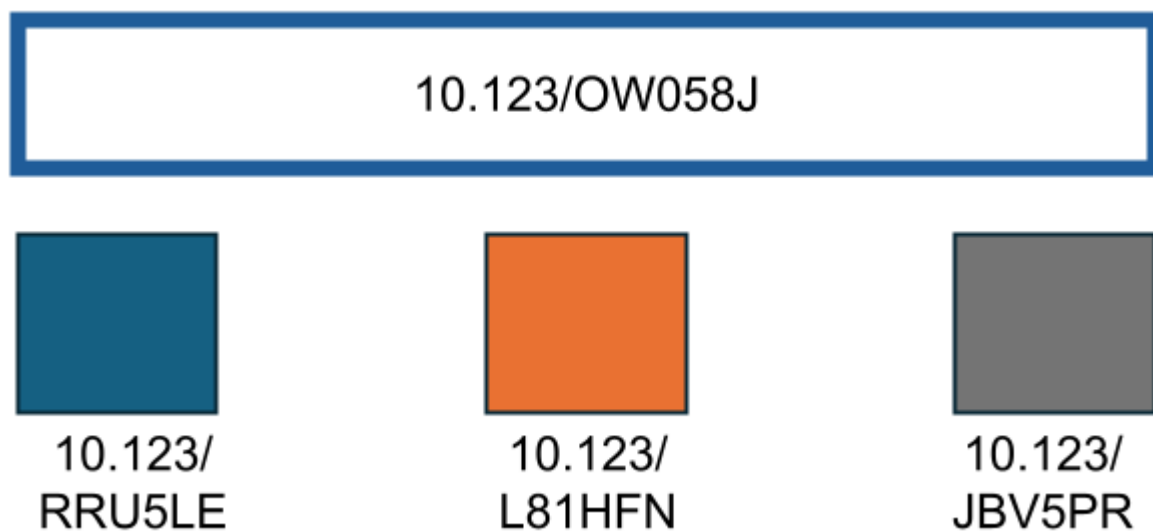


Figure 2: Collection of objects, where each object contained in the collection has its own identifier (Klump, 2024; Klump et al., 2016).

Time series

Time series are a special case of a changing data set in which changes occur only by appending new data (Figure 3). Identifying a time series as a whole may be sufficient, but there may also be use cases where snapshots of the time series are identified as data releases, e.g., monthly data releases.

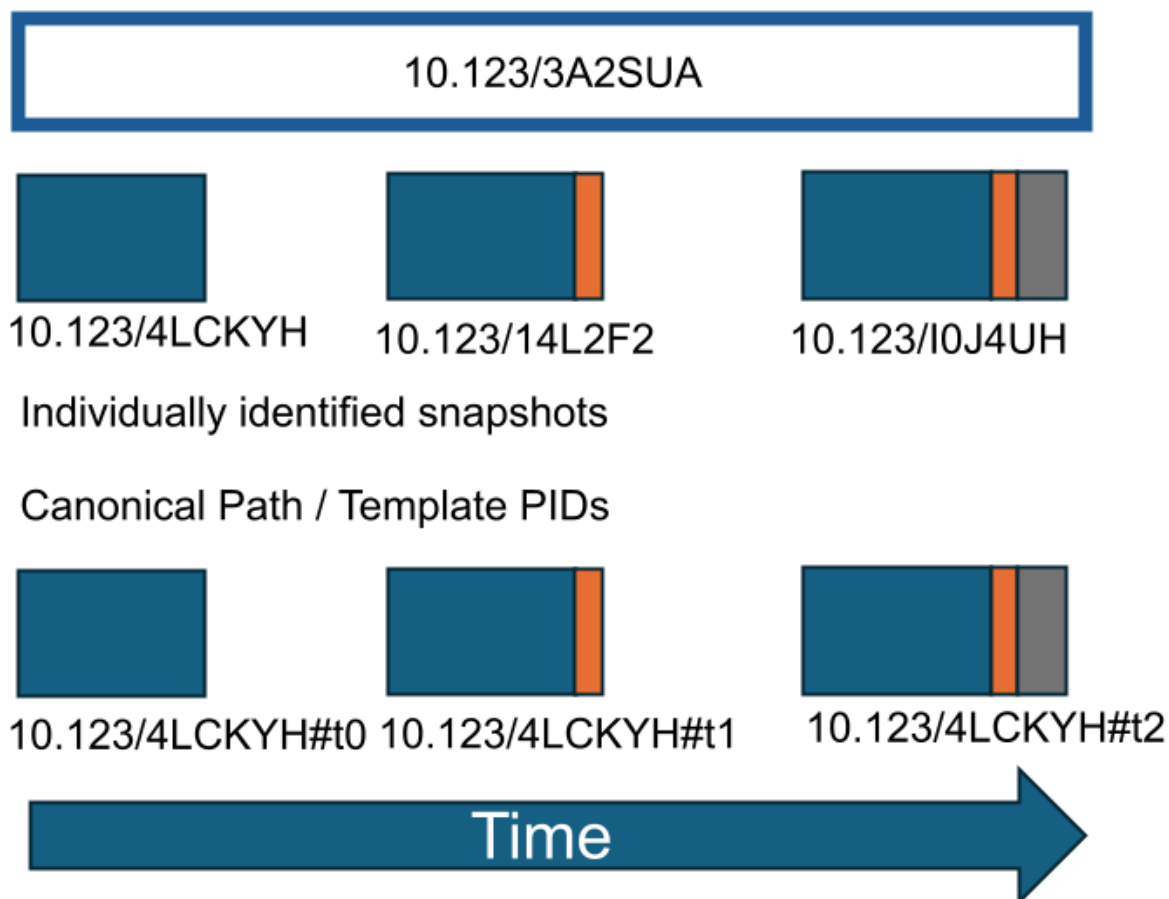


Figure 3: (top row) Identification of a time series as a whole with identified data snapshots at times T_0 , T_1 and T_2 as individually identified data releases and (bottom row) identification of subsets through a canonical path. (Klump, 2024)

Recommendations

- Collections: Identify the collection as a collection item. In the case of very large collections, i.e. thousands to millions of objects, persistent identifiers for individual items in the collection can be minted on demand.
- Time series: Time series in which data are only appended and not inserted can be identified by one persistent identifier. Subsets can be identified either by defining a canonical path or by using template PIDs.

Mirroring and re-publication

Sometimes, data items are mirrored to another platform. This is often done to provide easier access or greater bandwidth than the original data platform would be

able to offer. As an example, the hyperspectral scans of geological drill cores published by the Australian State and Territory Geological Surveys are available through the respective Geological Survey data portals and as a federated search through the National Virtual Core Library (NVCL, <https://www.auscope.org.au/nvcl>). To enhance performance and make large-scale data analysis of NVCL data easier, all hyperspectral data were mirrored to the high-performance computing platform at the National Computational Infrastructure (NCI, <https://nci.org.au/>).

Recently, data centres have been asked to demonstrate their relevance and impact to funding bodies. In the example described above, this would have led to a significant underestimate of the impact of the original data publications due to the larger number of downloads of the mirrored data from a platform with a better bandwidth availability for data transmission.

Cases like the NVCL data at NCI will become increasingly common, but no agreed-upon best practices exist for identifying and citing these kinds of data. For transparency, the original source of the data should be cited as the authoritative source (**authoritative version**) using the “IsIdenticalTo” attribute in the DataCite metadata. Unfortunately, the current version of the DataCite metadata (Version 4.5) (DataCite Metadata Working Group, 2024) does not provide means to identify which version is the authoritative and the mirrored version.

Recommendations

- Mirroring and re-publication: The mirrored or re-published data should always refer to the original source (authoritative version) using the “IsIdenticalTo” attribute in the DataCite metadata.

Deleted items and Retractions

Item deleted

Digital artefacts and the metadata about digital artefacts are usually stored independently. This includes the version history as part of the metadata. If the digital artefact is deleted or unavailable, the version history (consisting of changelogs, hash sums, etc.) should remain accessible.

If the object was identified by a persistent identifier, the identifier will persist and should still resolve to the metadata of the deleted object. The DataCite support pages comprehensively summarise best practices for this case (DataCite, 2024).

Metadata deleted

If the object was identified by a persistent identifier, the identifier will persist. If the metadata no longer exists, the identifier should resolve to a tombstone page providing further information.

Data publication retracted

Data publications that have been retracted still need to be identified and described. Whether the retracted data are still accessible will depend on the policy of the data repository. Retracted data can be kept accessible if this is necessary to support the reproducibility of results even if they are wrong. There might also be cases where retracted data are not made accessible to prevent misuse. In both cases, the retraction should be explained in the accompanying release notes.

Recommendations

- Data unavailable: A persistent identifier should always resolve to a landing page displaying the metadata, even when a dataset has become unavailable.
- Metadata unavailable: If both the data and the metadata have become unavailable, their persistent identifier should resolve to a tombstone page.
- Retractions: Display the metadata with an appropriate release note describing that the data have been retracted.

Provenance

Systematic versioning supports the recording of the provenance of data, as the version history and the **release notes** already record part of the provenance. In principle, the chain of data products derived from one another should be traceable. In the case of mirrored data, reference could be made to the source object (**authoritative version**).

Recommendations

- Release notes: The provenance of a data item should be recorded in its accompanying release notes.

- Git revision management: Using Git for revision management automatically records provenance. Most Git platforms also support release management with accompanying release notes.

Formats (Manifestations)

In software versioning, all items have the same format. Here, changes in the bitstream are always indicative of changes in content. This is not necessarily the case with data. Sometimes, the same data product is produced in different formats but with the same content. For example, a dataset might be available as character-separated values and in JSON format. In this example, the content is identical, but the encoding is different, resulting in bitstreams with different checksums. In another example, a geophysical map might be offered in netCDF format and as a GeoTIFF. In both cases, the content is the same on the level of the data product (**expression**), but its values are scaled differently on the data format level (**manifestation**).

There are no known examples where manifestations of data have been separately identified with PIDs.

Recommendations

- Format variants: The metadata of the data product (**expression**) should record different format variants (**manifestations**) of data. There are no known examples where manifestations of data have been separately identified with PIDs.

Citation

Data citation has been one of the primary motivations for developing formal data publication pathways (e.g. Altman and King, 2007; Costello, 2009; Klump et al., 2006; Parsons et al., 2010). Identification and versioning can help support citation, but the purpose of the citation has to be clear. Credit is given to the creator of this data product by citing the **expression**. Reference to the **item** supports reproducibility and gives credit to the infrastructure providing access to this resource.

The distinction of referencing the expression or the item leads to a problem of conflicting aims in the current system of citation metrics. Citing the data product (expression) will give credit to the authors and their research output but will not differentiate who provided the service of making the data accessible. Citing the item will give credit to the infrastructure providing access to the data product but might lead to skewed metrics in the cases of mirrored data (see section on mirrored data).

Recommendations

- Citation still follows the established practices for citation in scholarly communications but is not universally accepted by all journals.
- Credit is given to the creator of a data product by citing the **expression**.
- Citing the **item** will give credit to the infrastructure providing access to the data product but might lead to skewed metrics in the cases of mirrored data

Versioning of Metadata

Some self-service repositories like Figshare trigger a version change of the data upon changes to the metadata.

Should Metadata be identified and versioned? This question is currently under discussion. The versioning of metadata records was part of the STD-DOI metadata and DataCite metadata schema version 1.1, which was released in 2004 and 2010. This metadata element was dropped from the DataCite metadata schema 2.0 and subsequent releases because no compelling use case existed and to avoid confusion with the versioning information of the identified data object.

The rise of machine-readable metadata records ingested by metadata aggregators creates a new use case for metadata identification and versioning. The secondary publication of metadata by metadata aggregators creates a use case where human users and machines need to be able to identify different versions of a metadata record and, among those, the authoritative or any other specific source.

Recommendations

- While this use case may exist, there is no consensus on implementing metadata versioning in metadata aggregators. Also, care must be taken to communicate clearly to the users which identifier is used for the object described by the metadata and which identifier refers to the metadata record.

Versioning Systems

- Versioning practices need to be commensurate with the nature of the data and the stage in the data life cycle.

Keeping track of research materials can be quite challenging, especially in collaborative projects. By improving the ability to track and retrieve each version of a file consistently, we can enhance collaboration efficiency and ensure the accuracy of research results. This is best achieved through the use of version control systems that automate storage and record-keeping tasks.

Version control systems come in a wide range of complexities and features. This spectrum includes simple tools that automatically sync different versions of your files with different cloud storage solutions, as well as more complex systems like Git, which enable detailed management of multiple versions through branching and merging. It's important to consider what these tools can theoretically do and how those potential capabilities are applied in real-world scenarios.

When choosing a version control system for your research, the most important factor to consider is what can be consistently used in your current environment. If the tools you choose are too difficult to use or are not available on all platforms where research materials need to be managed, individual researchers may start using their own tools and systems, leading to the same kind of confusion you were trying to avoid. Existing familiarity with a specific tool, tool availability, and ease of setting up those tools on new devices are all important factors to consider when selecting a version control system.

Once you determine which systems can be integrated effectively into your research setting, it's important to consider a few more factors in your assessment. Beyond typical considerations for choosing software, such as cost, the size and engagement level of the user and support community, the degree of community governance, licensing, etc., there are specific inquiries you should make.

Questions to Ask

On which versioning system to choose, the Center for Open Science makes the following recommendations (Center for Open Science, 2023):

- How much complexity is functionally required by your specific tasks and workflows? For example, complex data processing workflows may benefit from more complex tools compared to tracking sequential versions of a data file during analysis.
- How many versions do you need to retain, and for how long? Can some of the revisions that led up to a prior release be deleted? Make sure that any limitations align with your research and archival needs and obligations.
- How much of your data workflow will this tool record once it is implemented? If your team only has a few team members interacting with the version management system, your version control system may end up with incomplete or inaccurate information. For example, if only one data manager enters all the files into the group version control tool, it may be difficult to determine who made specific changes and when. If each researcher entered their own changes directly, the version control tool would capture that information automatically.
- Does the tool support an editorial workflow that supports data releases through a data publication platform?

Recommendations

- **Use Version Control Systems:** Implement version control systems to consistently track and retrieve file versions, enhancing collaboration and research accuracy.
- **Match Complexity to Needs:** Choose a version control system that fits your project's complexity, from simple cloud sync tools to advanced systems like Git.
- **Ensure Usability and Accessibility:** Select a system that is easy to use, available on necessary platforms, and familiar to your team to avoid confusion.
- **Evaluate Integration and Support:** Consider factors like cost, community support, and ease of integration when choosing a version control system.
- **Focus on Practical Application:** Prioritize systems that work well in your real-world research environment over those with just theoretical capabilities.

Related Concepts

The work of the RDA Data Versioning IG/WG is based on other related concepts from software development, library and information science, and archiving, which the group adopted to support their work.

Functional Requirements for Bibliographic Records

The Functional Requirements for Bibliographic Records (FRBR) are an entity-relationship model that describes how data objects, actors, and organisations relate to each other and how these relationships can be described (Figure 4). FRBR was designed to support cataloguing and searching of multimedia content, it does not model a data life cycle.

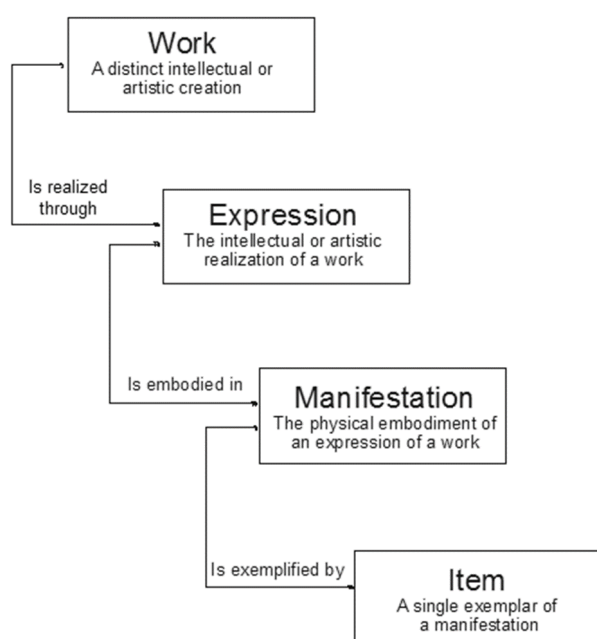


Figure 4: Schematic illustration of the FRBR model.

Figure 5 below illustrates the application of FRBR to data as proposed by the RDA Data Versioning IG/WG. In this example, the **work** is the Greenland Ice-Core Project, which has its **expression** in the paper published by (Dansgaard et al., 1993). The paper has several **manifestations**, among them the print version of the article, available as an **item** on the shelves of Library A. The oxygen isotope data measured on the GRIP ice core were **expressed** as a data product, which is **manifested** as CSV data and available for download as an **item** from PANGAEA (Johnsen, 1999) and the NOAA National Center for Environmental Information (Johnsen et al., 1997)

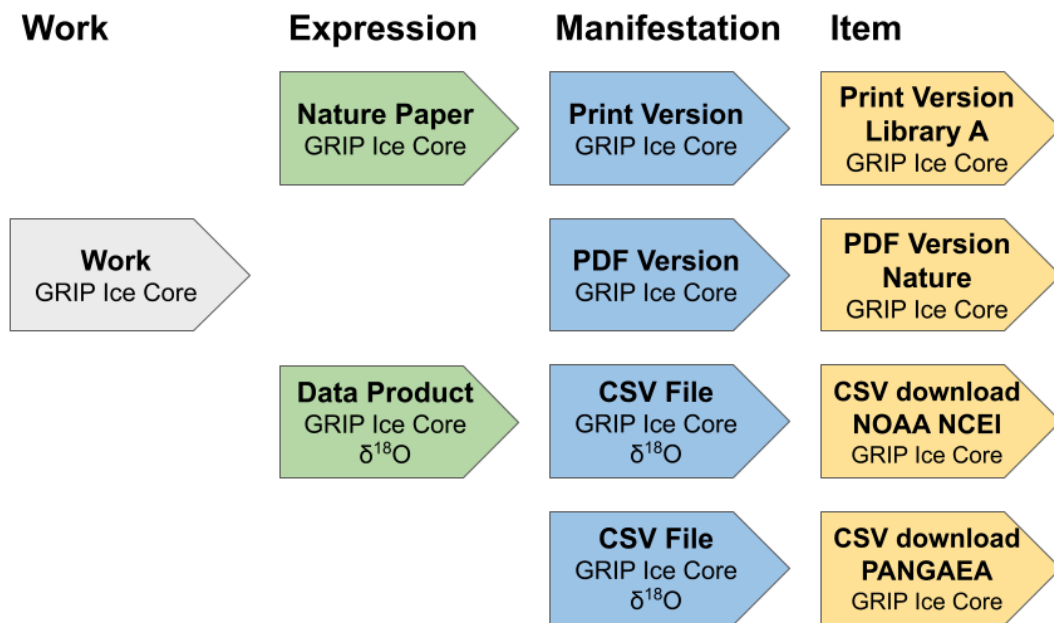
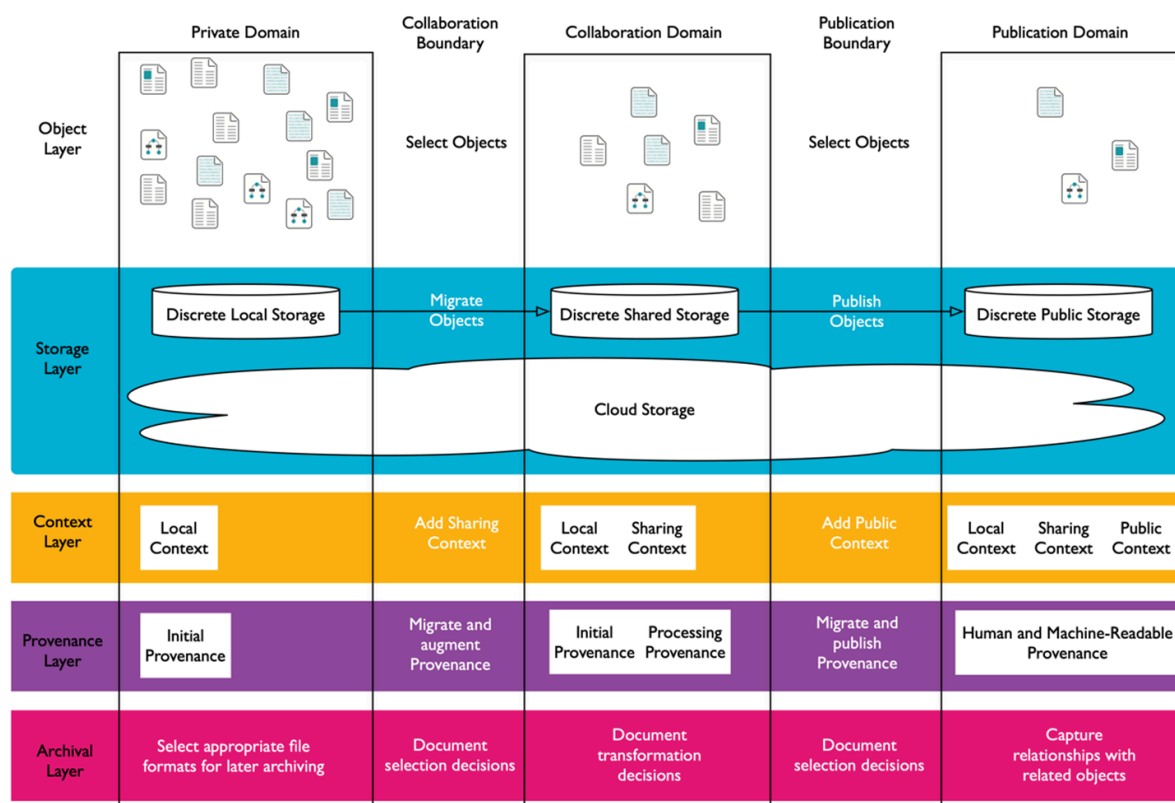


Figure 5: Application of the FRBR model to scientific publications and data using the interpretation of the Greenland Ice-Core Project (GRIP) (Dansgaard et al., 1993; Johnsen, 1999; Johnsen et al., 1997) as an example.

Data Life Cycle and Domains of Responsibility

Dataset versioning is always a part of the data life cycle. Since versioning should come with adequate versioning information, it is important to consider the respective stages of the data life cycle and the availability of implicit and explicit metadata. The concept of “Domains of Responsibility” (Treloar et al., 2007; Treloar and Klump, 2019) helps identify curatorial responsibilities and the availability of implicit and explicit metadata.

In a typical workflow, illustrated in Figure 6, data are generated in the Private Domain, shared in the Collaboration Domain and then published or archived in the Publication Domain. Access to published or archived data is through the Publication Domain—the metadata accompanying a dataset changes along this path.



Version 2.2, @atreloar, 3 Mar 2019
Research Objects image CC-BY <http://researchobject.org/>

Figure 6: Schematic view of the Domains of Responsibility. In a typical workflow, data are generated in the Private Domain, shared in the Collaboration Domain and then published or archived in the Publication Domain. Figure from (Treloar and Klump, 2019).

Private Domain

Most of the time, data are created in the Private Domain. In this domain, most metadata are an implicit part of the context of the Private Domain in which the dataset is created and with which the creator of the data is familiar. At this stage, there might be many data objects that were created in the course of research, but not all of them will be kept. Some will be shared with others in the Collaboration Domain.

Collaboration Domain

When data are shared with others in the context of a project or other collaboration, it enters the Collaboration Domain. Here, the context has to be made more explicit in the form of metadata. At the same time, some information about the shared data objects will be available to all collaborators as implicit information. Version control and keeping track of revisions are also most important in the private and collaboration domains.

Publication Domain

Upon publication, or for archiving data after the end of a project, data are transferred from the Collaboration to the Publication Domain. When a dataset is transferred to the Publication Domain, the curatorial responsibility is passed on from the individual researcher or research group to a memory institution, e.g. a data repository.

Because of the absence of the data creators in the Publication Domain, all contextual information is lost unless explicitly encoded in the data description, including metadata, release notes, and versioning information (Treloar and Klump, 2019).

Data can be retrieved from the Publication Domain for reuse, thus completing the data life cycle. This data life cycle model aligns with the Open Archival Information Systems Reference Model (OAIS) (CCSDS, 2012), with the OAIS model covering the Publication Domain.

Open Archival Information Systems

The Open Archival Information Systems Reference Model (OAIS) (CCSDS, 2012) is a general model for archival systems, not only digital. It describes the elements of an archival system, agents interacting with the system, and their interactions (Figure 7).

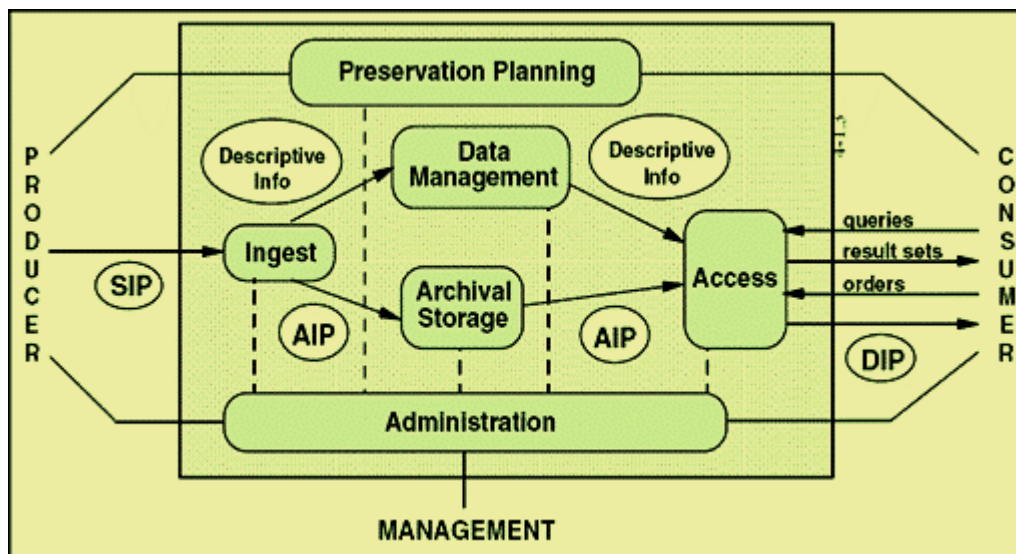


Figure 7: Elements of the OAIS Reference Model.

The OAIS Reference Model spans the Publication Domain in the model of (Treloar and Klump, 2019). It has an interface to the Collaboration Domain where Submission Information Packages (SIP) are ingested into the archival system. The objects are held in the archival system as Archival Information Packages (AIP). The format of the SIP and AIP do not need to be identical but are determined by the needs of the designated user community and archival processes. Data for reuse are disseminated through the access interface of the archival system as **items** and are served as Dissemination Information Packages (DIP). Again, the AIP and DIP do not need to be identical. The format of the DIP is determined by the needs of the designated

community, and the DIP may be made available in several formats (**manifestations**). The format might even change with time as requirements change.

Designated User Community

A key concept in the design of information systems is its **Designated Users**, which is prominently featured in OAIS (Bettivia, 2016; Parsons and Duerr, 2005). This concept is similar to the concept of a Persona in user experience design (Lidwell et al., 2023). It represents the goals and behaviour of a group of users rather than an individual and describes the role a person or organisation may take. Personas help guide the exploration of the context of behaviours and relationships between actors. In this sense, the Designated Users represent the roles in which organisations and people interact with the information system and their requirements.

With this concept from user experience design in mind, an information system should define its designated users when designing its services.

References

- Altman, M and King, G** 2007 A Proposed Standard for the Scholarly Citation of Quantitative Data. *D-Lib Magazine*, 13(3/4). DOI: <https://doi.org/10.1045/march2007-altman>
- Bettivia, R S** 2016 The power of imaginary users: Designated communities in the OAIS reference model. *Proceedings of the Association for Information Science and Technology*, 53(1): 1–9. DOI: <https://doi.org/10.1002/pra2.2016.14505301038>
- Buttigieg, P L, Cristiano, L, Curdt, C, Ihsan, A Z, Jejkal, T, Koch, C, Mannix, O, Mohr, D P, Pirogov, A, and Stucky, K-U** 2022 Guidance on Versioning of Digital Assets (Report No. HMC Paper 3). Kiel, Germany: HMC Office, GEOMAR Helmholtz Centre for Ocean Research. DOI: https://doi.org/10.3289/HMC_publ_04
- CCSDS** 2012 Reference Model for an Open Archival Information System (OAIS). Magenta Book (Recommended Practice No. CCSDS 650.0-M-2). Greenbelt, MD: Consultative Committee for Space Data Systems. Available at <http://public.ccsds.org/publications/archive/650x0m2.pdf>
- Center for Open Science** 2023 Version Control. *OSF Support*. Available at <https://help.osf.io/article/149-version-control> [Last accessed 24 June 2024].
- Costello, M J** 2009 Motivating Online Publication of Data. *BioScience*, 59(5): 418–427. DOI: <https://doi.org/10.1525/bio.2009.59.5.9>
- Dansgaard, W, Johnsen, S J, Clausen, H B, Dahl-Jansen, D, Gundestrup, N S, Hammer, C U, Hvindberg, C S, Steffensen, J P, Sveinbjörnsdóttir, A E, Jouzel, J, and Bond, G** 1993 Evidence for general instability of past climate from a 250-kyr ice-core record. *Nature*, 364(6434): 218–220. DOI: <https://doi.org/10.1038/364218a0>
- DataCite** 2024 Best Practices for Tombstone Pages. *DataCite Support*. Available at <https://support.datacite.org/docs/tombstone-pages> [Last accessed 9 July 2024].
- DataCite Metadata Working Group** 2024 DataCite Metadata Schema Documentation for the Publication and Citation of Research Data and Other Research Outputs v4.5 (Recommended Practice). Hannover, Germany: DataCite e.V. Available at <https://doi.org/10.14454/g8e5-6293> [Last accessed 26 January 2024].

Diepenbroek, M, Edmunds, R, Hugo, W, and Mokrane, M 2014 Guidelines in Respect of Metadata Granularity, Version 2.1 (Technical Report). Tokyo, Japan: World Data System. Available at <https://docs.google.com/file/d/0B4qnUFYMGSc-SFNORWFialhnZ3c/edit>

Hamming, R W 1950 Error detecting and error correcting codes. *The Bell System Technical Journal*, 29(2): 147–160. DOI: <https://doi.org/10.1002/j.1538-7305.1950.tb00463.x>

Jenkyns, R, Mathiak, B, McNeill, K, Smith, G, Sun, G, and RDA Data Granularity Working Group 2024 RDA Data Granularity WG Guidance Outputs (draft) (Draft for Public Comment). Research Data Alliance. Available at https://docs.google.com/document/u/1/d/1BRAkleKR7scvNlzkkeBpP-cZgduC88fMUk_R45p5t-o/edit?usp=drive_web&oid=103444092687085147709&usp=embed_face [Last accessed 31 May 2024].

Johnsen, S J 1999 GRIP Oxygen Isotopes. DOI: <https://doi.org/10.1594/PANGAEA.55091>

Johnsen, S J, Clausen, H B, Dansgaard, W, Gundestrup, N, Hammer, C U, Andersen, U, Andersen, K K, Hvidberg, C S, Dahl-Jensen, D, Steffensen, J P, Shoji, H, Sveinbjörnsdóttir, A E, White, J W C, Jouzel, J, and Fisher, D A 1997 NOAA/WDS Paleoclimatology - GRIP Ice Core 248KYr Oxygen Isotope Data. DOI: <https://doi.org/10.25921/M91S-K638>

Kálmán, T, Kurzawe, D, and Schwardmann, U 2012 European Persistent Identifier Consortium - PIDs für die Wissenschaft. In: Altenhöner, R and Oellers, C (eds.), *Langzeitarchivierung von Forschungsdaten - Standards Und Disziplinspezifische Lösungen*. Berlin, Germany: Scivero. pp. 151–168. Available at <http://www.ratswd.de/publikationen/langzeitarchivierung-von-forschungsdaten>

Klump, J 2024 *Workshop “Forschungsdaten-Publikationen zwischen Dynamik und Persistenz.”* DOI: <https://doi.org/10.5281/zenodo.12818783>

Klump, J, Bertelmann, R, Brase, J, Diepenbroek, M, Grobe, H, Höck, H, Lautenschlager, M, Schindler, U, Sens, I, and Wächter, J 2006 Data publication in the Open Access Initiative. *Data Science Journal*, 5: 79–83. DOI: <https://doi.org/10.2481/dsj.5.79>

Klump, J, Huber, R, and Diepenbroek, M 2016 DOI for geoscience data - how early practices shape present perceptions. *Earth Science Informatics*, 9(1): 123–136. DOI: <https://doi.org/10.1007/s12145-015-0231-5>

Klump, J, Wyborn, L A I, Downs, R R, Asmi, A, Wu, M, Ryder, G, and Martin, J 2020a Compilation of Data Versioning Use cases from the RDA Data Versioning Working Group. Research Data Alliance. Available at <https://doi.org/10.15497/RDA00041> [Last accessed 24 January 2020].

Klump, J, Wyborn, L A I, Wu, M, Downs, R R, Asmi, A, Ryder, G, and Martin, J 2020b Final Report of the Research Data Alliance Data Versioning Working Group - Principles and best practices in data versioning for all data sets big and small (Working Group Final Report). Kensington WA, Australia: Research Data Alliance. Available at <https://doi.org/10.15497/RDA00042>

Klump, J, Wyborn, L A I, Wu, M, Martin, J, Downs, R R, and Asmi, A 2021 Versioning Data Is About More than Revisions: A Conceptual Framework and Proposed Principles. *Data Science Journal*, 20(1): 12 p. DOI: <https://doi.org/10.5334/dsj-2021-012>

Lacan, O 2023 Keep a Changelog V1.1.1. *Keep a Changelog*. Available at <https://keepachangelog.com/en/1.1.0/> [Last accessed 19 June 2024].

Lidwell, W, Holden, K, and Butler, J 2023 *Universal Principles of Design, Updated*

and Expanded Third Edition: 200 Ways to Increase Appeal, Enhance Usability, Influence Perception, and Make ... Decisions. Expanded 3rd edition. Beverly, MA: Rockport Publishers.

Martinez, C 2017 Reproducible Research Version Control. *Data Carpentries*. Available at <https://datacarpentry.org/rr-version-control/> [Last accessed 10 May 2024].

O'Neill, E T 2002 FRBR: Functional Requirements for Bibliographic Records. *Library Resources & Technical Services*, 46(4): 150–159. DOI: <https://doi.org/10.5860/lrts.46n4.150>

Parsons, M A and **Duerr, R E** 2005 Designating user communities for scientific data: challenges and solutions. *Data Science Journal*, 4: 31–38. DOI: <https://doi.org/10.2481/dsj.4.31>

Parsons, M A, Duerr, R, and **Minster, J-B** 2010 Data Citation and Peer Review. *EOS, Transactions, American Geophysical Union*, 91(34): 297–298. DOI: <https://doi.org/10.1029/2010EO340001>

Rauber, A, Asmi, A, van Uitvanck, D, and **Pröll, S** 2015 Data Citation of Evolving Data - Recommendations of the Working Group on Data Citation (WGDC). Troy, NY: Research Data Alliance. Available at <https://rd-alliance.org/group/data-citation-wg/outcomes/data-citation-recommendation.html>

Rauber, A, Gößwein, B, Zwölf, C M, Schubert, C, Wörister, F, Duncan, J, Flicker, K, Zettsu, K, Meixner, K, McIntosh, L D, Jenkyns, R, Pröll, S, Miksa, T, and **Parsons, M A** 2021 Precisely and Persistently Identifying and Citing Arbitrary Subsets of Dynamic Data. *Harvard Data Science Review*, 3(4): 29 pp. DOI: <https://doi.org/10.1162/99608f92.be565013>

Software versioning 2019. *Wikipedia*. Available at https://en.wikipedia.org/w/index.php?title=Software_versioning&oldid=886437916 [Last accessed 11 March 2019].

Treloar, A E, Groenewegen, D, and **Harboe-Ree, C** 2007 The Data Curation Continuum - Managing Data Objects in Institutional Repositories. *D-Lib Magazine*, 13(9/10): 13. DOI: <https://doi.org/10.1045/september2007-treloar>

Treloar, A E and **Klump, J** 2019 Updating the Data Curation Continuum: not just Data, still focused on Curation, more about Domains. *International Journal of Digital Curation*, 14(1): 87–101. DOI: <https://doi.org/10.2218/ijdc.v14i1.643>