



Planning for Data Management

With some best practice tips

Jennifer Abel, University of Calgary
RDM Jumpstart
May 12, 2025

Why you need to plan for data management



- Research doesn't happen in a vacuum!
- There are many, many factors that will affect both a project overall and the data within a project
- If you don't plan for these factors before you start, you can run into big problems as you go along
- Even without big problems, planning to manage your data will make your life a lot easier

The key areas you need to plan for



1. Ethical, legal and commercial issues
2. Data collection
3. Data documentation
4. Storing, accessing and working with data
5. Specifics of procedures/workflows
6. Long-term data management, discoverability and access

Ethical, legal and commercial issues

1. Ethical, Legal and Commercial Issues



It's possible you'll work on a project where none of these are relevant to your work. However, if you do research with any of the following, they can be crucial:

- Human participants in research (e.g., health research, behavioural research, surveys, user testing...) or personal information/personal health information
- Animals
- Researchers at other institutions, inside or outside of Canada
- Industry partners or community organizations (e.g., using a product that a company developed; surveying community service users)
- Data that someone else collected
- Indigenous communities and/or Traditional Knowledge
- Areas of research that a government has said are sensitive

1. Ethical, Legal and Commercial Issues



What you need to consider:

- Laws (e.g., privacy laws, copyright laws, laws around access to health data)
- Legal agreements and frameworks (e.g., contracts with partners, licenses for products or data, copyright)
- Ethical frameworks (for working with humans or animals)
- Federal or provincial policies (e.g., around research security)
- Funder policies (e.g., around sharing data)
- Institutional policies, procedures and standards (e.g., around privacy, intellectual property, cybersecurity, submitting theses/dissertations)
 - Any specific guidelines around the student/supervisor relationship at your institution

1. Ethical, Legal and Commercial Issues



You need to plan to meet whatever obligations you're under. E.g.,

- How you'll ensure that data are safely and securely stored during the active phases of the project
- Who'll have access to the data
- What will happen to the data after the project is complete
 - Will you share any of it? If so, how and where, and what permissions/licenses will you apply?
 - Do you have to keep any of it for a particular period of time?
 - Will any of it have to be destroyed?

Sidebar: Assessing the Risk Level of Data




When you're planning, it's useful (and often essential) to assess the risk level of your data.

Ask yourself, "What would happen if someone other than a member of the research team had access to the data?"

Will release of any or all of the data:

- harm my data sources?
- make me or my institution liable?
- adversely impact my collaborators?
- adversely impact by ability to share my findings?
- pose a security threat?

Example: Risk Categories for Human Participant Data

	Low Risk	Medium Risk	High Risk	Extreme Risk
Risk Level Definitions	Publicly available data where there is no reasonable expectation of privacy, regardless of sensitivity or identifiability.	All identifiers collected have been stripped so that data to be deposited has no information that could reasonably identify individuals or groups.	Identifiers remain and/or (re)-identification is possible or probable. Data subjects may be vulnerable in the context of the research and may be harmed if a breach were to occur.	Data acquired through an agreement (formal or informal) with a custodian, barring further use or retention.

Abbreviated form of:

Sensitive Data Expert Group. (2020). Sensitive Data Toolkit for Researchers Part 2: Human Participant Research Data Risk Matrix. Zenodo. <https://doi.org/10.5281/zenodo.4060449>

Example: Federal Research Security Guidelines

Certain categories of research are deemed sensitive by the Canadian government, e.g.,

- Critical minerals or infrastructure
- Large datasets that could be sensitive
- Dual-use research with civilian and military potential

If your research falls into one of these categories, you need to fill out a risk assessment form before you can get federal funding:

<https://science.gc.ca/site/science/en/safeguarding-your-research/guidelines-and-tools-implement-research-security/national-security-guidelines-research-partnerships/national-security-guidelines-research-partnerships-risk-assessment-form>

Data Collection

2. Data collection



This concerns all the data you'll collect, observe, generate, and/or acquire.

Consider:

- How you'll get it (i.e., how you'll collect, observe, generate and/or acquire it)
- What kind(s) of data will be involved (e.g., textual, numeric, images, video, audio...)
- What format(s) the data will be in (e.g., .docx, .txt, .xlsx, .csv...)
- How much data will be involved (Kilobytes? Megabytes? Gigabytes? Terabytes?)
- Whether any of it is subject to any of the concerns in the previous area

A note on file formats: Proprietary vs. non-proprietary

Proprietary

Require specialized software/hardware to open/read

Might not be openable/readable if the software/hardware ceases to exist

May be industry standard

Examples: Microsoft Word files (.docx),
Microsoft Excel files (.xlsx)

Non-proprietary

Readable by various kinds of software/hardware, including open source software

Generally good for making data FAIR

May be industry standard

Examples: Plain text files (.txt), comma-separated values files (.csv)

A note on versions of data



Important when you're thinking about how much storage you need

- Raw: what you obtained directly from your research
- Master (sometimes called processed data): have been manipulated to remove errors/outliers, prepare data for analysis, derive new variables, de-identify participants, etc.
- Analytic: the files you do the analysis work on
- Analyzed: the results of qualitative or quantitative analysis (often presented as charts, tables, graphs, etc.)
- Final: master/processed data in a preservation-friendly format

Data

Documentation

3. Data Documentation



Think about the procedures/documents you'll use to make sure that your data are easily read and interpreted correctly throughout the research process, including after project completion.

Some examples of documentation:

- Research methodologies
- Code (used in conducting analyses)
- Standard operating procedures
- [Data dictionaries](#)
- [Codebooks](#)
- [Readme](#) files

Data dictionaries, codebooks and readmes



“Data dictionary” and “codebook” are often used interchangeably: they’re both documents that outline the layout, structure, content, and meaning of the variables in a dataset. (see <https://www.nlm.gov/guides/data-glossary/data-dictionary> ; <https://guides.library.upenn.edu/c.php?g=564157&p=9554907>)

Readmes provide information about a data file/dataset and are intended to help ensure that the data can be correctly interpreted by yourself or others (see e.g., <https://data.research.cornell.edu/data-management/sharing/readme/>)

Storing, Accessing and Working with Data

4. Storing, accessing and working with data



You need to think about both where and how this will happen, during the active phases of the project.

Consider:

- The versions of data that will be worked with (raw, master, analytic)
- The activities involved (collection, processing, analysis, dissemination)
- The software and platforms you'll be using to work with the data
- Who needs access, and any necessary security measures
- How data will be backed up to prevent data loss

Notes on storage and backup



- Your institution may have rules/guidelines on where you can store research data during a project
 - Check with IT
- If you can, follow the 3-2-1 rule:
 - 3 copies of your data, on 2 different storage media, with 1 copy in a physically different location from the others

Specifics of Procedures/ Workflows

5. Specifics of procedures/workflows

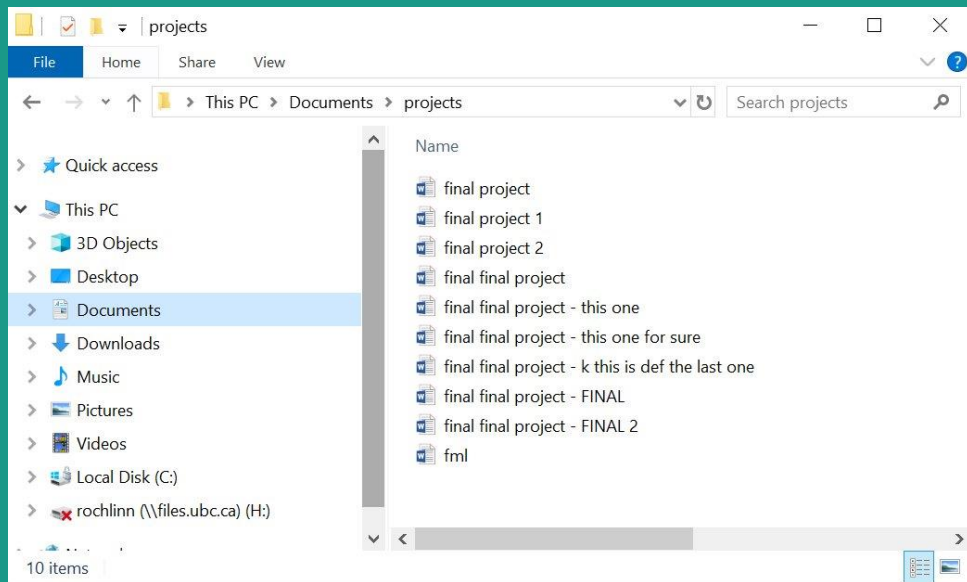


Consider things like


- How you're naming your files
- What your folder hierarchy will be
- How you'll clean your data
- How often things are backed up
- Who can change/edit documentation

Also consider who's responsible for doing these things, and what resources will be needed


A Specific Specific: File Naming Best Practices




File Naming Best Practices

- 
- Human readable
 - Machine readable
 - Consistency!


Human Readable

- 
- Can you look at a file name and know what it is? What about in a year from now?
 - Will others be able to look at your files and know what they are?
 - Will you/others be able to easily find a file that you/they are looking for?

Human Readable

- 
- Short but complete names
 - Ideally 3-5 conceptual elements
 - Write down your naming conventions in a README file
 - Define acronyms, abbreviations, codes, etc.

Human Readable

- 
- Short but complete names
 - Ideally 3-5 conceptual elements
 - Write down your naming conventions in a README file
 - Define acronyms, abbreviations, codes, etc.

Elements to consider in file naming:

- Date of creation/collection
- Group/affiliation
- Activity
- Location
- Editor/creator
- Version
- Other relevant information

Example: Data Files




Example: `lldr_mpp_2025-04-22.csv`


Documentation:

- Convention: description_location_collection-date.file-type
- lldr: leaf litter decomposition rate
- mpp: Monk Provincial Park


Machine Readable

- 
- How will a computer parse your file names?
 - If a file moves from one computer / application / operating system to another, will they remain interpretable in the same way?

Machine Readable - Best Practices

- 
- Only contain letters in the English alphabet, numbers 0-9, dashes -, and underscores _
 - Do not use spaces or special characters such as ~!@\$%^&*()_+{}|
 - Separate naming elements with underscores and dashes
 - Use date format: YYYYMMDD or YYYY-MM-DD

Machine Readable - Best Practices

- 
- Only contain letters in the English alphabet, numbers 0-9, dashes -, and underscores _
 - Do not use spaces or special characters such as ~!@#\$%^&*()+{}|
 - Separate naming elements with underscores and dashes
 - Use date format: YYYYMMDD or YYYY-MM-DD

Does anybody know why these are considered best practices?

Machine Readable - Ordering

- A big part of organizing files has to do with how machines order characters.
- There can be subtle differences across operating systems and applications that can be quite complicated as is beyond this series, but the following can be good general guidelines to follow for interoperable ordering:
 - Ordering begins with the first character of a file name, and works its way from left to right.
 - Numbers are ordered ahead of alphabetical characters.
 - Dashes - are ordered ahead of underscores _, and both are ordered ahead of numbers.
 - While some systems will position capital letters ahead of lowercase, it is not recommended to use letter casing as a way to order names.

Machine Ordering - Test Yourself!



How would the following file names be ordered?

- 1) session-10_file-naming.pptx
- 2) session-1_intro-to-rdm.pptx
- 3) -README.txt
- 4) session-10_file-naming.docx

Machine Ordering - Test Yourself!



How would the following file names be ordered?

- 1) -README.txt
- 2) *session-9_intro-to-osf.pptx
- 3) session-10_file-naming.docx
- 4) session-10_file-naming.pptx

*May not always happen! Some systems will prioritize the first number it sees, so 10 may appear higher than 9 because the system is looking at 1 vs. 9. To avoid this, you may want to consider giving numbers in file names a numerical system like 01, 02... or 001, 002...

Likeness and Importance

- When choosing file names, consider which elements are the most important, and how likeness/differences with play into how the names are sorted.
- Which element should come first? Second? Third? ...

Elements to consider in file naming:

- Date of creation/collection
- Group/affiliation
- Activity
- Location
- Editor/creator
- Version
- Other relevant information


Version Control

- Version control refers to the systematic tracking of the various versions and growth of your files.
- There are 3 main ways this can be handled:
 - Manual systems
 - Automated systems
 - Scripting


Version Control - Manual Systems

- Manual version control is a way to track files via file naming, and lends itself well to administrative documents and manuscripts, but in the right context can be appropriate for data as well.
- Examples of manual version control:
 - Version number: manuscript_v01.docx, manuscript_v02.docx, ...
 - Editor initials: manuscript_v01_NR.docx
 - Stage/process of data: data_raw.csv, data_clean.csv
- For the purpose of the Jumpstart, we will not be using manual version control, but it's worth knowing it exists as it may be valuable in your own research.
- Please ask if you're interested in knowing more!


Version Control - Automated Systems

- 
- Automated version control systems are computer applications that employ metadata, similar to manual systems, that don't require manual input.
 - There are many different systems that can handle automated version control, and they vary widely in their abilities and complexity.
 - For the purpose of the Jumpstart, we will be using OSF to handle some level of automated version control, but if you are interested in other systems, please ask!


Version Control - Scripting

- 
- Scripting, which will be covered in detail over days 2-4, is a way to handle version control by providing a systematic list of all activities that were performed on a data file or files.
 - It allows both you and others to be able to review and re-perform any changes or actions, and can be used to save files periodically (with file naming to denote different stages of the data).
 - For the purposes of the Jumpstart, this will be our primary means of version control, but as you may see, there is generally crossover across version control systems.

Directory Structures

- 
- Directories, AKA folders, are a way of keeping your files organized and easy to find.
 - The same principles for file naming apply to directories.
 - Developing a directory structure before you begin a project can help with managing all the files that will be collected or generated.

An Example

-  Mountain Legacy Project, 2024, "MountainScape Segmentation Dataset", <https://doi.org/10.5683/SP3/CEYU10>, Borealis, V2

File Naming for the Week

- For the purpose of this workshop series, you will be working with 2 directories to save your files:
 - Scripts → a place to save your code files
 - Data → a place to save your data files
- You won't be working with too many files this week, so you're not at risk of making a huge mess, and you are free to name your files how you see fit.
- With that said, you can use this as an opportunity to play around with file naming conventions and to think about how things might work if there were many more files to deal with.

Questions on file naming?

Long-term data
management,
discoverability and
access

6. Long-term data management, discoverability and access

What happens to your data after the project is complete

Consider:

- What data you'll keep, what you'll destroy, and what you'll share
- What you'll need to do to allow you to keep/destroy/share the data
- All the software and platforms you'll need to allow you to do this (e.g., long-term storage options, data repositories)

6. Long-term data management, discoverability and access

Also consider:

- What do your funders/publishers/regulators/partners/supervisors require you to do with the data? (see part 1)
- What scripts/software/code/metadata are necessary to allow continued access to/usability of the data? (see part 3)
- If you're sharing data, how will you do that? What do you need to put in place at the beginning of your project so you can do that? (e.g., ethics approval, agreement with your supervisor, contracts, deciding on a data repository)

A preliminary note on data repositories



Data repositories are one way that you can share your data.

- Online database services that provide long-term preservation for data and make them available for discovery and use
- Used after a project is complete and data won't be changing on a regular basis
- Intended for sharing data, rather than just keeping it but having it inaccessible

We'll go into more detail on repositories on Day 5.

- Make sure you sign up for a Borealis Demo account!

Questions?
