Best Practices in Research Data Management

Hauke Sonnenberg & Michael Rustler

2018-06-21 09:01:11

Contents

Preface

> "These days, data trails are often a morass of separate data and results and code files in which no one knows which results were derived from which raw data using which code files."
>
> — Professor Charles Randy Gallistel, Rutgers University

This document is the outcome of the KWB project FAKIN (Forschungsdatenmanagement an kleinen Instituten = research data management at small institutes).

It defines best practices for research data management. It is mainly based on the personal experiences of

the authors having worked in many different research projects at KWB.

The document is outlined as follows:

- Chapter 1
- Chapter 2 defines Best Practices for different topics.
- Chapter 3 gives overviews about KWB projects and lists how different data related tasks have been solved within these projects and
- Chapter 4 provides FAQs.

This document is assumed to be a "living" document. We highly appreciate any comments and suggestions for improvements. What are your experiences with research data managment tasks? Can you provide solutions for specific tasks?

Example blocks:

TIP

NOTE

WARNING

CAUTION

IMPORTANT

Chapter 1

Introduction

1.1 Why RDM ?

Doing research data management (RDM) is useful according to Labfolder.com (2018) as it:

- Saves time for research
- Avoids risk of data loss
- Ensures transparency and reproducibility
- Increases data visibility and number of citations
- Fulfill funders' requirements and receive more grants
- Produce new knowledge and make more discoveries just by re-using data
- Archive, retrieve and re-use own data

1.2 For whom ?

Info on the document – create links between projects and thematic groups

Chapter 2

Best Practices

2.1 Plan and fund a new project

2.1.1 Data management plan

Action 1: Look through a data management planning checklist DMP checklist?

Action 2: Create a data management plan. You can use DMP Online as it has a template or if required, use a funder's format.

RDM checklist before starting with a new project Brauchen wir eine Data Management Plan? See e.g. https://dmponline.dcc.ac.uk/

2.1.2 Electronic Lab Notebooks

Electronic Lab Notebooks (ELNs) https://www.labfolder.com/research-data-management/ ELNs are an essential asset for researchers to fulfil any requirements for data management, and they create direct bridges between scientists and stakeholders. By adopting an ELN, the data lifecycle can proceed smoothly and easily: from creating and collecting data digitally in one place to one-click data archiving, ELNs empower researchers by allowing them to implement their RDM plan without effort and time investment.

An ELN to make your data FAIR? Welches ist das?

2.1.2.1 Project acronyms

At the start of a project we define a project acronym. This acronym is intended to be used in file and folder names.

Whenever we want to indicate the relation to a certain project in a file or folder name, we use the project acronym in exactly the typing that was defined. This is important as we want to distinguish between raw data, processed data and project results in our Folder Structures.

The project acronyms are defined in a simple text file `PROJECTS.txt` in the `//server/projects$` folder, see Project Folder Structure.

2.2 Data storage

2.2.1 Folder structure

@Michael: Regeln für die Verbesserung der Ordnerstruktur @Michael: Vorschlag für verbesserte Ordnerstrukturen

2.2.2 Naming

@Michael: Konventionen für Ordner- und Dateinamen, Vorschlag einer Nomenklatur @Michael: Regeln zur Vermeidung langer Dateipfade

Useful file names are:

- consistent
- meaningful to you and your colleagues
- allow you to find the file easily

It is useful if your department/project agrees on the following elements of a file name:

- Vocabulary – choose a standard vocabulary for file names, so that everyone uses a common language
- Punctuation – decide on conventions for if and when to use punctuation symbols, capitals, hyphens and spaces
- Dates – agree on a logical use of dates so that they display chronologically i.e. YYYY-MM-DD
- Order - confirm which element should go first, so that files on the same theme are listed together and can therefore be found easily

• Numbers – specify the amount of digits that will be used in numbering so that files are listed numerically e.g. 01, 002, etc.

!!!!! Good practice: Remove spaces from file names or use punctuation such as underscores and hyphens to separate words e.g. “AHRC_TechnicalApp_Response20120925.docx” or “AHRC-TechnicalApp-Response20120925.docx” rather than “what we got back from funders about the data stuff.docx !!!!

2.2.3 Raw data

Especially in environmental sciences, raw data often cannot be reproduced (e.g. rainfall, river discharge measurements) and are therefore of high value.

- dürfen umbenannt werden, dies muss in den Metadaten dokumentiert werden
- dürfen inhaltlich nicht verändert werden
- werden schreibgeschützt abgelegt
- werden in einem eigenen Bereich abgelegt (ein Ordner enthält alle Rohdaten)

2.2.4 Versioning

Versioning or version control is the way by which different versions and drafts of a document (or file or record or dataset or software code) are managed. Versioning involves the process of naming and distinguishing between a series of draft documents that leads to a final or approved version in the end. Versioning allows you to disclose an audit trail for the revision and update of drafts and final versions.

2.2.4.1 Manual

We propose the following workflow:

- It is only allowed to modify the current file (e.g. `filename.pptx`), which contains no version name as postfix
- A version is created by copying the current file. This copy gets a not already defined version name as file name ending (e.g. `filename_v0.1.pptx`)
- The versioned file is moved to a folder 'VERSIONS" and set as read-only.
- In the folder `VERSIONS` is a file `VERSIONS.txt`, which contrains addtional information on the available version within that folder

Drawbacks:

- possibly more time demanding,
- needs time getting used to it and
- requires high level of discipline

Advantages:

- Simple and safe method, which requires not version control software
- Cleaner workspace as old versions are stored in a separate folder
- Sorting by name equals the chronology

Example:

```
BestPractices_Workshop.ppt
VERSIONS/
  + VERSIONS.txt
  + BestPractices_Workshop_v0.1.ppt
  + BestPractices_Workshop_v0.2.ppt
  + BestPractices_Workshop_v1.0.ppt
```

Content of file `VERSIONS.txt`:

```
BestPractices_Workshop.ppt
- v0.1: first draft version, written by Michael Rustler
- v0.2: after additions by Hauke Sonnenberg
- v1.0: first final version, after review by Pascale Rouault
```

2.2.4.2 Automatically

The versioning is done automatically in case a version control software, like Git or Subversion are used.

At KWB we currently use the following version controll software:

- Subversion: for KWB r. Requires connection to the KWB intranet
- Git: for publishing programm code external on our KWB organisation group on Github. Currently all repositories are public (i.e. are visible for everyone), but also use of private repositories is possible for free as KWB is recognised as non-for-profit company by Github, offering additional benefits for free

Use of version control software is required in case of programming (e.g. in R, Python, and so on) and can be useful in case of tracking changes in small text files (e.g. configuration files that run a specific R script with different parameters for scenario analysis).

Drawbacks:

- Special software (TortoiseSVN), login data for each user on KWB-Server and some basic training are required
- In case of collaborate coding: sticking to 'best-practices' for using version control is mandatory, e.g.:
  - timely check in of code changes to the central server,
  - Speaking to each other: so that not two people work at the same time at the same program code in one script as this leads to conflicts that need to be resolved manually, which can be quite time demanding. You are much better off if you avoid this in the upfront by talking to each other

Advantages:

- Only one filename per script (file history and code changes are managed either internally on a KWB server in case of using TortoiseSVN or externally for code hosted on Github)
- Old versions of scripts can be restored easily
- Additional comments during `commit` (i.e. at the time of transfering the code from the local computer to the central version control system about `why` code changes were made and build-in diff-tools for tracking changes improve the reproducibility

Attention: version contral software is not designed for versioning of raw data and thus should not be used for it. General thoughts on the topic of 'data versioning' are available here: https://github.com/leeper/data-versioning

2.2.5 Organising your e-mail account

From: https://www.data.cam.ac.uk/data-management-guide/organising-your-data#Refernces Most people now routinely send and receive lots of messages every day and as a esult, their inbox can get very quickly overloaded with hundreds of personal and work-related email. Setting aside some time to organise your emails will ensure information can be found quickly and easily, and is stored securely. Why should I organise my email? Apart from the obvious frustration and time wasted looking for that email you remember sending to someone last month, email is increasingly used to store important documents and data, often with information related to the attachments within the email itself. How can I ensure my emails remain organised?

Here are some general tips to ensure your email remains organised in the long term:

- Delete emails you do not need. Remove any trivial or old messages from your inbox and sent items on a regular (ideally daily) basis.
- Use folders to store messages. Establish a structured file directory by subject, activity or project.
- Separate personal emails. Set up a separate folder for these. Ideally, you should not receive any personal emails to your work email account.
- Limit the use of attachments. Use alternative and more secure methods to exchange data where possible (see 'data sharing' for options). If attachments are used, exercise version control and save important attachments to other places, such as a network drive.

2.2.6 Managing references

Refer to KWB Endnote guidelines....

2.2.7 Metadata

What information would someone need to find/re-use your data? Location, title, creator name, description, date collected

- keep metadata in plain text file “readme.txt”

Tools for metadata tracking and data standards are:

- Metadatenstandards prüfen, z. B. DataCite (siehe u.a. ZALF, GFZ Potsdam)

2.2.8 Data preservation

Data must be retained to support your research findings

Standards: ?5 years?

2.3 Data collection

Data are often inconsistent, incomplete, incorrect, or misspelled Data cleaning is essential You may also use OpenRefine http://openrefine.org/ to clean your messy data Or use the following tools

2.3.1 Logger devices

2.3.2 Spreadsheets

Chapter 3

Projects

In general the projects can be divided into two groups:

- Case studies: tested in detail within FAKIN (i.e. proposed best-practices will be applied for this case studies and cross-checked whether their application is useful
- Others: just suming up the data workflows and created tools for enlarging the KWB internal knowledge what has already been done.

3.1 Case studies

3.1.1 Geogenic salination

3.1.2 Modelling LCA modelling

Umberto

3.1.3 Pilot plants (AQUANES) (#aquanes)

Created R package:

- aquanes.report: data import, temporal aggregation, interactive visualisation of operational data of pilot facilities and joining with lab data

Sites (among others):

- Berlin-Tiefwerder
- Berlin-Schönerlinde
- Basel Lange-Erlen
- Haridwar

Challenges:

- zeitlich hoch aufgelöste Betriebsdaten (~ 10 Mio. Datenpunkte pro Monat) erforderten massive Performance Optimierung um die Visualisierung der Rohdaten im Tool über den Testbetriebszeitraum (18 Monate) auf Rechnern mit limitierten RAM Ressourcen (~ 8 GB) zu ermöglichen.
- Nutzung: wird von den Projektpartnern zum strukturierten Datenimport genutzt um dann darauf ggf. eigene weitere Analysen darauf aufzusetzen. Für die beiden Berliner Standorte wird zudem die

Visualiserung der Betriebsdaten routinenäßig genutzt, um vom Regelbetrieb abweichende Zustände besser indentifizieren zu können.

3.2 Others

3.2.1 Spree2011 (2007)

Used data by source (data formats in parentheses)

- KWB:
  - water level and discharge at one monitoring site (Text/CSV)
  - rain (Text/CSV)
- BWB:
  - pumping rates in the pumping stations (Excel)
  - water levels in the pumping stations (Excel)
  - rain at some gauges near the monitoring site (Excel)

Tasks and methods by topic

- Dry-weather and wet-weather calibration of a sewer network model (Infoworks)
  - InfoWorks: Creating rain input files
  - InfoWorks: Creating RTC input files

Questions that arose:

- Where to store presentations (trainee vs. employee)?
- Where to store the raw data (personal drive of the trainee)?
- How does Infoworks interpret timestamps, how do BWB provide timestamps? -> metadata

3.2.2 MIACSO (2009)

Monitoring

- sites: one site in the sewer (monitoring container), more sites in the river
- variables: water quantity and quality
- devices: online sensors

Modelling

- Sewerage: Infoworks
- River hydraulics: Hydrax
- River quality: QSim

Data storage

- High amount of data -> extra server: `moby`
- We put some effort in planning good folder structures for the data. Nevertheless the structure at the end of the project is not as clean as it was planned.
- Data that we received from project partners was stored in `Daten/EXTERN`.
- Raw data was stored in a folder `Daten/RAW` which was write-protected and required a special user-account for storing new data.

```
Daten/
  ACCESS/ # MS Access databases, containing raw data
  EXTERN/ # External data (by organisation)
  META/   # MS Access databases, containing metadata
```

```
        # (about calibration, maintenane, sites, variables)
RAW/    # Text files containing raw data, from KWB-own devices only, by site
```

Metadata

- many devices in the container -> meta data about device cleaning and maintenance important -> tool: `META_Maintenance.mdb`

Methods and Tools

- We imported most of the data from text files into MS Access databases in -> tool: `MiaCsoRawImport.mdb`
- We calibrated the sensors offline by using SQL queries to provide calibrated data from raw data -> tool: `MiaCsoMetaCalibControl.mdb`
- We used SQL queries to perform data processing -> tool: `MiaCsoStatAnalysis.mdb`
- Data validation (outlier detection) was done in a two step procedure:
  1. Automatic preselection using MS Access tool `MiaCsoStatAnalysis.mdb`
  2. Manual selection using self-developed graphical tool in Origin

Developed Tools:

- MS Access Applications
  - `MetaMaint.mdb`: Monitoring Metadata Management
  - `MiaCsoRawImport.mdb`: Text File Import to MS Access
  - `MiaCsoStatAnalysis.mdb` (project deliverable): Definition and automatic execution of sequences of SQL queries
- Origin extension to interactively select and store outliers graphically
- R packages
  - `kwb.mia.evalCritO2` (project deliverable): graphical evaluation of critical oxygen conditions in the river
  - `kwb.mia.iw`: Calculation of file sizes of InfoWorks result csv-files exported from InfoWorks.
  - `kwb.miacso`: functions used in MIA-CSO, for example for plotting data availabilities.

3.2.3 KURAS

Developed Tools:

- Frontend for KURAS Database of Rainwater Management Measures: `KURAS_DB_Acc2003_hs.mdb`
- R package `kwb.kuras`: Interface to KURAS database

3.2.4 OGRE

- Decision to use CUAHSI Community Observations Data Model (ODM)
- R script to import lab data from Excel to MS Access database implementing ODM

Developed Tools:

- R packages
  - `kwb.ogre`
  - `kwb.ogre.model`
  - `kwb.odm`
  - `kwb.odmx`

3.2.5 Flusshygiene

- Adaptation of free online monitoring data visualisation HydroServerLite
- Reusage of lab data import script developed in OGRE

3.2.6 DEMOWARE

Entstandene R Pakete:

- Grundwassermodellierung
  - kwb.hantush
  - kwb.vs2dh
  - kwb.demoware
- Quantitatives mikrobiologisches Risikomanagement
  - kwb.qmra: wird im Rahmen von AQUANES(#aquanes) weiter genutzt

3.2.7 OPTIWELLS

Created R packages:

- kwb.wtaq: Groundwater Modelling
- kwb.epanet: (Pressure)Pipe Network Simulation (EPANET)

3.2.8 RWE

(Semi)automatisierte Erstellung eines komplexen MODFLOW Modells (mehrere Layer, mehrere hunderte Entnahmebrunnen mit zeitlich variierender Entnahmemenge sowie Hinzufügen/Entfernen von Brunnen innerhalb des Simulationszeitraums) in Phyton mittels flopy sowie Entwicklung der Modellszenarien auf Github (siehe hier: Maxflow).

Input Christian !?!

Chapter 4

FAQs

4.1 What is digital curation?

Digital curation is the selection, preservation, maintenance, collection and archiving of digital assets.