

1.1 Dateibenennung und Versionierung

Daten managen – 1 Datenstrukturierung

Wir stellen uns vor, eine Versorgungsforscherin arbeitet mit mehreren Leuten an einem Forschungsprojekt. Jede Person recherchiert und erarbeitet Daten und Informationen aus verschiedenen Quellen zu unterschiedlichen Teilbereichen: Es werden Übersichtsdocuments geschrieben, Tabellen zusammengestellt, Auswertungen angefertigt, ... Der gemeinsame Dateiordner wächst immer weiter:

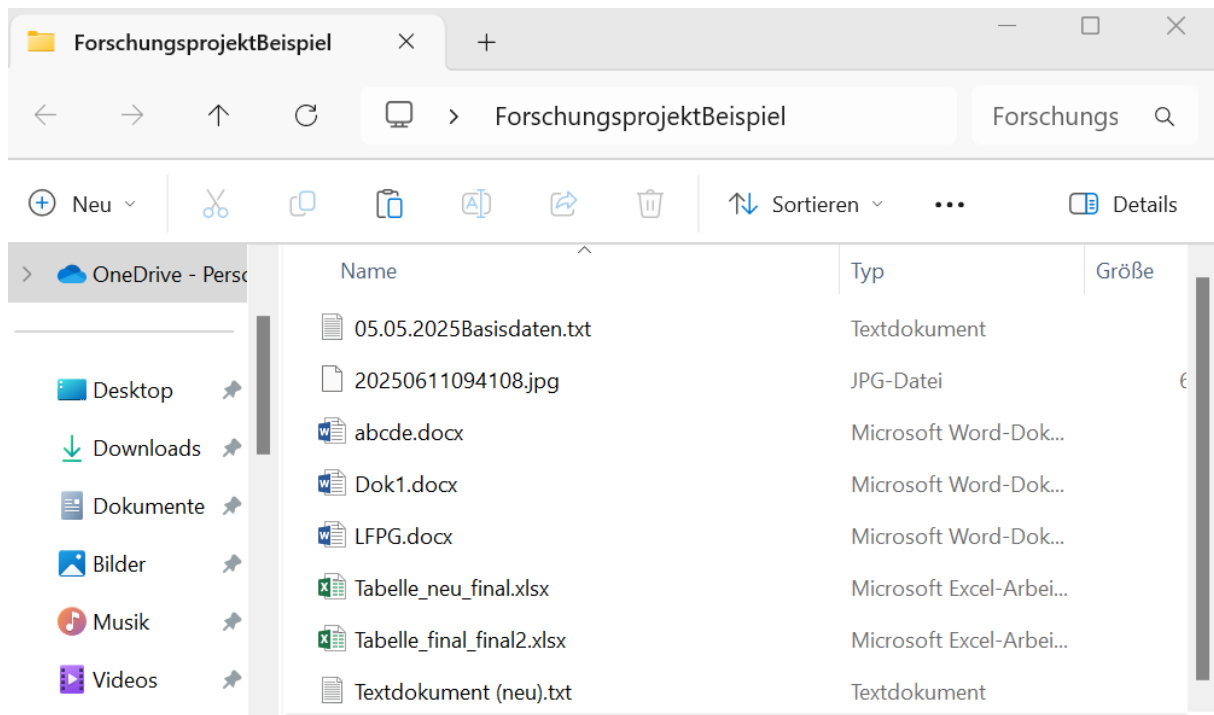


Abb. 1. Negativbeispiel für die Dateibenennung

Beim Betrachten der Dokumente kommen ihr folgende Gedanken in den Kopf:

- Was stand nochmal in dem Textdokument?
- Was ist das nochmal für ein Bild?
- Hm, die Benennung war eine Abkürzung für irgendwas, aber ich kann mich nicht erinnern?
- Welche Tabelle war jetzt wirklich die finale?

Sie muss die Dateien ständig öffnen, um die Inhalte zu prüfen, teilweise hilft auch das nicht wirklich weiter. Ständig fragen sie ihre Kolleg:innen, wo sie bestimmte Informationen finden. Das Projekt schreitet immer weiter fort und so auch die Frustration im Team. Eine bessere Benennung der Dateien muss her!

Dateinamen

Dateibenennungen sollten logisch nachvollziehbar sein und auch von fremden Personen verstanden werden können. Folgende Faktoren sollten bei der Entwicklung einer einheitlichen Benennung beachtet werden:

1. Das **System** unter dem die Datei gespeichert ist (manche Systeme erlauben z.B. keine Sonderzeichen oder nur Benennungen bis zu einer bestimmten Namenslänge).

2. Der **Kontext**, denn unabhängig vom Speicherort muss durch inhaltsspezifische und deskriptive Informationen erkennbar sein, zu welchem Kontext die Datei gehört (z.B. ZeitplanProjektname statt nur Zeitplan).
3. Die **Konsistenz**, um Einheitlichkeit der Benennung zu erreichen. Namenskonventionen sollten in einer ReadMe-Datei festgehalten werden. Enthalten sein sollten die Erläuterung von Abkürzungen, die Reihenfolge der Benennung und die enthaltenen Informationen.

Bestandteile von Dateinamen

Es gibt einige Standard-Bestandteile, die sich zur Benennung von Dateien anbieten, allerdings keine ultimative „One-size-fits-all“-Lösung. Praktikable Benennungen können sich je nach Fachausrichtung, Anzahl der Zugriffspersonen, individuellen Vorlieben, etc. unterscheiden. Folgende Beispiele sind in viele Fälle praktikabel:

Bestandteil	Beschreibung	Beispiel
Datum	Erstellungs-/Änderungsdatum der Datei (bestenfalls im JJJJMMTT-Format)	20250326
Inhaltsstichwort	Deskriptive Information, kurz und prägnant	Auswertungsprotokoll
Typisierungskennung	Beschreibt Dokument-, Daten-, Medientyp	int (Interview), note (notizen)
Identifikationskennung (ID)	Eindeutige fortlaufende Kennzahl (z.B. für Interviewpartner:innen)	001, 002, ...
Person	Name oder Namenskürzel (der Urheber:in, Interviewperson); Achtung bei Initialen die doppelt Vorkommen	MM (oder Max Mustermann)
Versionskennung	Angabe von Bearbeitungsversionen	v1_02
Segmentierung (Dateiendung)	Benennung der Teile eines Ganzen Bezeichnet das Dateiformat der Datei (kann vom System automatisch angezeigt werden)	001_a, 001_b .csv, .docx

Falls man gleichzeitig mehrere Dateien umbenennen will gibt es dafür spezielle Software-Tools, wie [PowerRename](#) (für Windows) und [Renamer](#) (aktuell Version 7 für Mac). Linux besitzt dafür integrierte Funktionen.

Benennungskonventionen

Eine gute Dateibenennung sollte sowohl für Mensch und Computer verständlich und nutzbar sein. Deswegen haben sich folgende Konventionen bei der Benennung etabliert:

- „So kurz wie möglich und so lang wie nötig“: Als Faustregel sollten Dateinamen maximal 32 Zeichen haben. Die meisten Systeme erlauben maximal eine gesamte Pfadlänge von 255 Zeichen.
- Der Name sollte deskriptive Informationen bieten, die Rückschlüsse auf den Inhalt der Datei erlauben.
- Sonderzeichen, Umlaute und Leerzeichen sollten vermieden werden. Manche Zeichen werden systemseitig sogar nicht erlaubt. Einzige Ausnahme: Der Punkt am Ende zur Abtrennung der Dateiendung.
- Wenn die Benennung aus mehreren Worten/Teilen besteht: Worttrennung erfolgt am besten durch CamelCaseBenennung (neues Wort beginnt mit Großbuchstaben) oder

snake_case_benennung (neues Wort wird mit _ abgetrennt).

- Eine Datumsangabe sollte am Anfang erfolgen, um die Sortierungsfunktion des Systems ausnutzen zu können. Für eine chronologische Sortierung empfiehlt sich das JJJJMMTT-Format.
- Automatisch generierte Dateinamen sollten immer umbenannt werden z.B. die typischen (und unverständlichen) Zahlenfolgen bei Fotos.
- Wenn Dateien nummeriert werden sollen muss die Skalierbarkeit beachtet werden. Eine zweistellige Dateinummer erlaubt beispielsweise nur Daten von 00 bis 99. Falls mehr Dateien erwartet werden sollten direkt die notwendigen Stellen durch führende Nullen freigehalten werden.
- Da die Namenslänge begrenzt ist, sind sinnvolle Kürzel oft zu empfehlen. Diese sollten in einer ReadMe-Datei hinterlegt werden.
- Bei mehreren Iterationen von Dateien existieren, sollte Versionierung verwendet werden

Versionierung

Versionen helfen einen Überblick zu behalten, in welcher Reihenfolge die Dateien im Bearbeitungsprozess entstanden sind und die Prozessschritte zu gruppieren. Eine ergänzende Prozesshistorie kann die Veränderungen transparent festhalten. Je nach Bedarfen des Projekts gibt es unterschiedliche Möglichkeiten:

Die gängigste Form, um Versionen zu kennzeichnen, ist die Vergabe von ganzen Zahlen für größere Veränderungen und einer kleineren Zahl dahinter für kleinere Veränderungen (z.B. v1, v1_02 etc.). Von Bezeichnungen wie final, final2, revision, definitiv_final, neu,... wird abgeraten!

Also, besser:

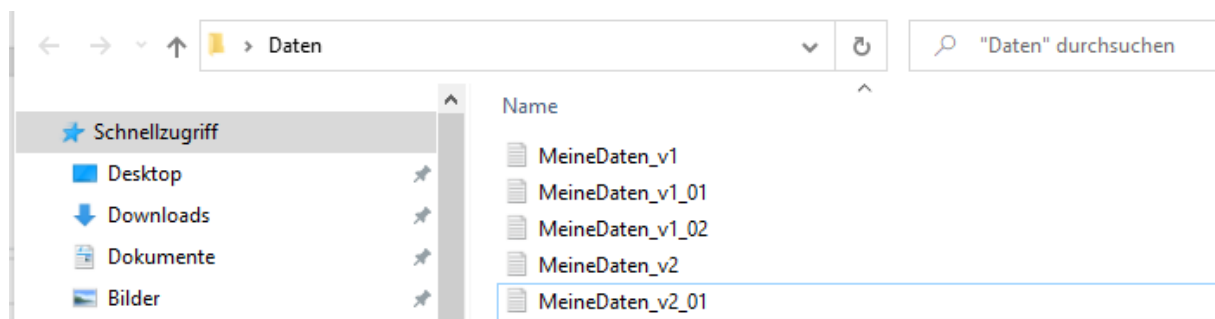


Abb. 2. Positivbeispiel für die Versionskennzeichnung von Dateien

Eine andere Möglichkeit ist eine dreistufige Benennung: Die **erste Stelle** wird verändert, wenn Fälle, Variablen, Wellen, Samples oder andere große Veränderungen hinzugefügt werden. Die **zweite Stelle**, wenn Daten korrigiert werden, so dass Analysen und Ergebnisse beeinflusst werden. Die **dritte Stelle**, wenn einfache Überarbeitungen ohne Bedeutungsrelevanz vorgenommen werden.

Weiterführend kann auch eine Versionskontrollsoftware wie [Git](#) oder [TortoiseSVN](#) verwendet werden, um Versionen zu verwalten. Diese bieten sich insbesondere für kollaboratives Arbeiten an. Bei Versionskontrollsoftware ist es möglich, verschiedene Entwicklungszweige zu erstellen, was die Verwaltung von alten oder alternativen Versionen erleichtert. Diese Entwicklungszweige können auch später wieder verschmolzen werden. Auch hier sind sinnvolle Ordner- und Dateistrukturen essenziell wichtig. Zusätzlich können die Anpassungen in den Versionen in einem Changelog

Gefördert durch:



Bundesministerium
für Forschung, Technologie
und Raumfahrt



Finanziert von der
Europäischen Union
NextGenerationEU

(„Änderungsprotokoll“) dokumentiert werden. Bei Versionskontrollsoftware werden diese Informationen aktiv eingefordert:

- Was wurde geändert?
- Wann wurde es geändert?
- Warum waren die Änderungen notwendig?
- Wer ist verantwortlich?

Quellenverzeichnis

Blümm, M., Fritsch, K., Bock, S., Hackenbuchner, J., Arning, U., & Förstner, K. U. (2024). 07_LE_Dateistruktur. FDM@Studium.nrw Blended-Learning-Basiskurs „Forschungsdatenmanagement“ (Version 1.0). Abgerufen am 30.09.2025, von https://landesinitiativefdmnrw.github.io/FDMatStudium/thk/texte/07_LE_Dateistruktur.html

Krähwinkel, E., Langner, P., Lipp, R., & Pietsch, A. M. (2022, März 22). HeFDI Data OER: Forschungsdatenmanagement – eine Online-Einführung. Zenodo. <https://doi.org/10.5281/zenodo.6373596>

Mertzen, D., Neuroth, H., Schneemann, C., Woywod, K., Haase, C., Jacob, B., Kroehling, M., Mittelbach, J., Straka, J., Szczukowski, A., & Weise, K. (2023, Mai 27). Zertifikatskurs "Forschungsdatenmanagement für Studierende": Spring School 2023 der Landesinitiative für Forschungsdatenmanagement in Brandenburg. Zenodo. <https://doi.org/10.5281/zenodo.7936966>

Weiterführende Info

MIT Libraries Data Management Group (2016). Research Data Management: Strategies for Data Sharing and Storage. Abgerufen am 30.09.2025, von <https://www.twillo.de/edu-sharing/components/render/01887843-1814-42e6-9ad5-6941a88d5c5a>

Software Carpentry (2018). File Organization: Naming. Abgerufen: 30.09.2025, von <https://datacarpentry.org/rr-organization1/01-file-naming/index.html>

Das Material ist im Rahmen des Projekts „DIM.RUHR: Datenkompetenzzentrum für die interprofessionelle Nutzung von Gesundheitsdaten in der Metropole Ruhr“ (Förderkennzeichen: 16DKZ2008[A-F]) entstanden. Dementsprechend spiegelt das Material in Version 1.0 den Stand von 2025 wider.

„Daten managen - 1.1 Datenstrukturierung – Datenbenennung und Versionierung“ von Anne Mainz und Sven Meister.

Dieses Werk und dessen Inhalt sind – sofern nicht anders angegeben – lizenziert unter CC BY 4.0. Ausgenommen aus der Lizenz sind die verwendeten Logos.

Der Lizenzvertrag ist hier abrufbar:

<https://creativecommons.org/licenses/by/4.0/>

