

Forschungsdatenmanagement

Notwendige, aber nicht hinreichende Voraussetzung
für den wissenschaftlichen Erkenntnisgewinn

18. Metadaten während der Datenerhebung

Dr. habil. Till Biskup

Physikalische Chemie

Universität Rostock

05.07.2024





- ❏ Daten ohne Kontext (Metadaten) sind wertlos.
Die Information *während* der Datenerhebung ist maximal.
- ❏ Eine Formalisierung der Metadatenaufnahme ist essentiell für Vollständigkeit und reale Umsetzbarkeit.
- ❏ Metadaten sollten maschinenlesbar vorliegen, um bei der Datenverarbeitung genutzt zu werden.
- ❏ Ein Dateiformat für Metadaten muss menschengeschrieben sein, ohne die Maschinenlesbarkeit zu korrumpieren.
- ❏ Kernaspekte sind Modularität und Erweiterbarkeit, wesentliche Felder Zweck und Kommentar.

Problemstellung, Herausforderungen und Anforderungen

Konzept und Umsetzung

Anwendung auf die eigene Praxis

Vorteile gegenüber Alternativen

Problemstellung

- ▶ relevante Metadaten *während* der Datenaufnahme möglichst vollständig erheben
- ▶ Die Information *während* der Datenerhebung ist maximal.

Antwort auf sechs Fragen

- ❓ Wer hat
- ❓ was
- ❓ mit wem
- ❓ wann
- ❓ wie
- ❓ und warum gemacht?

Herausforderungen

- ▶ Aufzeichnung aller relevanten Metadaten während der Datenerfassung in einer maschinenlesbaren Form
- ▶ Unabhängig von Dateiformaten, Betriebssystemen, und Netzwerkinfrastruktur (einschließlich Internetzugang)

Anforderungen

- ▶ Vom Menschen schreibbares und maschinenlesbares Format, mit Schwerpunkt auf der menschlichen Seite
- ▶ Ausreichend einfach und elegant in der Anwendung: sollte sich gut für einen Studenten im Grundstudium eignen
- ▶ Passt sich gut an die sich ändernden Anforderungen der Wissenschaft an: robust, modular, flexibel und (leicht) erweiterbar

Kontext

- ▶ wissenschaftlicher Kernaspekt
 - Nachvollziehbarkeit
- ▶ Station im Forschungsdatenlebenszyklus
 - Datenerhebung
- ▶ Bedeutung
 - Nachvollziehbarkeit der Datenerhebung
 - Ausgangspunkt für die Qualitätskontrolle und Qualitätssicherung

Beitrag

- ▶ Schema(ta) für Metadaten
 - inkl. Hinweisen und Heuristiken für eigene Schemata
- ▶ Dateiformat für die Metadatenablage

Problemstellung, Herausforderungen und Anforderungen

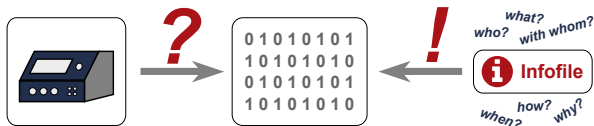
Konzept und Umsetzung

Anwendung auf die eigene Praxis

Vorteile gegenüber Alternativen

Towards more reproducible and FAIRer research data: documenting provenance during data acquisition using the Infofile format

*Bernd Paulus, Till Biskup**



Digital Discovery 2:234–244, 2023

- ▶ Ausgangspunkt: „Liesmich“-Datei eines Chemotechnikers
 - elektronisches Laborbuch direkt bei den Daten
 - Beobachtungen und Informationen zur Messung

Kriterien

- ▶ maschinenlesbar und vom Menschen les- und schreibbar
 - Fokus: Les- und *Schreibbarkeit* durch den Menschen
 - Voraussetzung: (einfache) Parsbarkeit
- ▶ Reintext (ASCII-7-Bit-Zeichensatz)
 - Grund: mangelnde Unterstützung von Unicode in MATLAB
 - Beschränkung für aktuelle Nutzung nicht mehr notwendig
- ▶ eindeutige Identifizierbarkeit des Dateiformats
 - durch eine Kennung in der ersten Zeile
 - inkl. Versionsnummer

Listing 1: Minimalbeispiel einer Infodatei

```
common Info file - v. 0.1.0
```

GENERAL

```
Date start: 2020-04-04
Time start: 11:05:00
Date end:   2020-04-04
Time end:   15:50:00
Operator:   Hans Wurst
Purpose:    Zeit totschiagen
```

SAMPLE

```
Name:      Zufallsprobe 1
Description: Vom Nachbarn gemopst
```

COMMENT

```
Sein oder nicht sein...
```

- ? Wer hat
- ? was
- ? mit wem
- ? wann
- ? wie
- ? und warum gemacht?

- ▶ Parameter als Schlüssel–Wert–Paare
- ▶ gruppiert in (logische) Blöcke
- ☞ Viele weitere Blöcke/Parameter für spezifische Methoden

- ▶ allgemein
 - Blöcke mit Schlüssel-Wert-Paaren
 - erste Zeile zur Identifizierung von Format und Version
- ▶ Feldnamen
 - werden mit Doppelpunkt vom Feldinhalt getrennt
 - dürfen keine Doppelpunkte im Namen selbst tragen
 - Leerzeichen sind erlaubt
 - müssen mit einem Buchstaben beginnen
- ▶ Leerraum (Leerzeichen, Tabulator, ...)
 - wird ignoriert, wenn er nach Feldnamen und vor und nach Feldinhalten auftritt
 - Feldinhalte *können* vertikal ausgerichtet werden.

- ▶ Zeilenumbrüche in Feldern
 - Folgezeilen müssen mit Leerraum beginnen.
 - ▶ Kommentare auf einer Zeile
 - Kommentarzeichen: %
 - Alles hinter dem Kommentarzeichen wird ignoriert.
 - Kommentarzeichen selbst durch Backslash verwendbar
 - ▶ Der letzte Block ist für Kommentare vorbehalten.
 - Im Kommentar ist jegliche Formatierung erlaubt.
 - Alles ab der Blocküberschrift bis zum Dateiende wird als Kommentar interpretiert und nicht geparkt.
 - Wiederkehrende Informationen sollten als neue Felder in die Infodatei aufgenommen werden.
- ☞ offenes, flexibles, modular erweiterbares Format

Listing 2: Infodatei mit Zeilenkommentaren

```
common Info file - v. 0.1.0

GENERAL
Date start: 2020-04-04
Time start: 11:05:00
Date end: 2020-04-04
Time end: 15:50:00
Operator: Hans Wurst
Purpose: Zeit totschiagen % Wichtig, nicht vergessen!

% In spezifischeren Formaten von Infodateien sollte ein Feld
% mit einem PID zur eindeutigen Referenz auf die Probe existieren.
% Weitere Informationen zur Probe dann z.B. in einem LIMS.
SAMPLE
Name: Zufallsprobe 1
Description: 50\% Glycerin im Puffer

COMMENT
Sein oder nicht sein...

Ab hier ist dann aber alles erlaubt, % inklusive Kommentarzeichen.
%% Das wird genauso gelesen wie es hier steht und nicht weiter verarbeitet.
Das ist der Grund, warum der Kommentar der letzte Block ist.
```

- ▶ Datum und Uhrzeit für Beginn und Ende angeben
 - Messungen über den Tageswechsel bzw. mehr als 24 h
- ▶ Name des/der Messenden angeben
 - bei mehreren Namen durch Komma getrennt
- ▶ Zweck einer Messung angeben
 - hilfreich beim Verstehen einer Messung
bzw. beim Wiederfinden des richtigen Datensatzes
- ▶ Proben mit eindeutigem Bezeichner (ID) versehen
 - im einfachsten Fall eine (fortlaufende) Nummer
 - lokale eindeutige und dauerhafte Kennung (UID, PID)
- ▶ Werte mit Einheit
 - Größe und Einheit durch Leerzeichen getrennt
 - sinnvollerweise Beschränkung auf SI-Einheiten

- ▶ reine Textdatei, von Menschen (einfach) schreibbar
 - keine externen Abhängigkeiten
- ▶ liegt bei/neben den Daten
 - unabhängig vom Dateiformat des Geräteherstellers
 - Daten und Metadaten sind immer beisammen
- ▶ maschinenverarbeitbare Metadaten
 - Auswertungsroutinen bekommen *semantische* Informationen und damit so etwas wie ein „Verständnis“ der Daten.
- ▶ hinreichend detailliert
 - nie wieder ein wichtiges Detail vergessen
- ▶ modular und erweiterbar
 - einfach an spezifische Bedarfe anpassbar

Problemstellung, Herausforderungen und Anforderungen

Konzept und Umsetzung

Anwendung auf die eigene Praxis

Vorteile gegenüber Alternativen

- ▶ relevante Metadaten herausfinden
 - Material- und Methodenteil einer Publikation sollte aus den Informationen in der Infodatei vollständig erzeugbar sein
 - Datenerhebung sollte mit der vorhandenen Information wiederholbar sein (unabhängig von der Verfügbarkeit der Probe)
 - Tipp: auf technisches Personal mit langjähriger Erfahrung hören und sie ermuntern, ihre Erfahrungen zu verbalisieren.
 - ▶ Heuristiken
 - Aufbauten aus unabhängigen bzw. austauschbaren Komponenten: jede Komponente hinreichend detailliert beschreiben
 - Geräte: Hersteller, Typenbezeichnung
 - Steuersoftware: Hersteller, Name, Versionsnummer, ggf. Datum
 - Kommentar für alle möglicherweise relevanten Zusatzinformationen, die (noch) kein eigenes Feld haben
- ☞ Setzt Erfahrung und gute Beobachtungsgabe voraus.

Blöcke für cw-EPR-Spektroskopie (Auswahl)

EXPERIMENT

Type: field-sweep
Runs: 1
Variable parameter: field
Increment: 0.02 mT

SPECTROMETER

Model: Bruker EMX
Software: Xenon 1.3b.5

MAGNETIC FIELD

Field probe type: Hall
Field probe model: xxx
Start: 341.4 mT
Stop: 361.5 mT
Step: 0.02 mT
Sequence: up
Controller: Bruker EMX
Power supply: Bruker ER 081

BRIDGE

Model: Bruker EMX PremiumX
Controller: Bruker EMX
Attenuation: 20 dB
Power: 2.00 mW
Detection: diode
Frequency counter: Bruker
MW frequency: 9.846977 GHz
Q value: 8300

PROBEHEAD

Type: HQ
Model: Bruker 4119HS-W1
Coupling: critical

TEMPERATURE

Temperature: 120 K
Controller: Oxford ITC 503
Cryostat: Oxford ESR935
Cryogen: LN2

- ▶ Schema für die eigene Anwendung entwickeln
 - Gruppierung zusammengehörender Information in Blöcke
 - Anordnung der Blöcke in einer logischen/intuitiven Reihenfolge (beim Einlesen egal, aber nicht unbedingt beim Ausfüllen)
 - Versionierung: Schemata entwickeln sich (gerade am Anfang)
 - Kritische Reflexion und Weiterentwicklung:
Welche Informationen tauchen immer wieder im Kommentar auf?
- ▶ Verwendung obligatorisch machen
 - wesentliche Komponente bei der Einweisung in die Methodik
- ▶ Datenverarbeitung, die die Informationen einliest
 - semantische Informationen, „Verständnis“ der Daten
 - Referenzimplementierung für den Infofile-Parser in Python
 - Format hinreichend einfach und eindeutig spezifiziert:
Parser-Implementierung in anderen Sprachen ist einfach.

Problemstellung, Herausforderungen und Anforderungen

Konzept und Umsetzung

Anwendung auf die eigene Praxis

Vorteile gegenüber Alternativen

Dateiformat

- ▶ intuitiv
 - (noch) einfacher vom Menschen schreibbar als YAML
- ▶ robust
 - Leerzeichen und Einrückungen sind weitestgehend optional
- ▶ resilient
 - Vorlage kann ausgedruckt und manuell ausgefüllt werden
 - Datei kann ausgedruckt und ins Laborbuch geklebt werden
- ▶ plattformunabhängig
 - reine Textdatei
 - Erstellen und Bearbeiten benötigt keine zusätzliche Software
- ▶ mögliche alternative Formate
 - XML, JSON, YAML
 - Gegenüberstellung auf den folgenden Seiten

Listing 3: Infodatei im XML-Format

```
<?xml version="1.0" encoding="UTF-8" ?>
<root>
  <file>
    <type>common Info file</type>
    <version>v. 0.1.0</version>
  </file>
  <general>
    <start>2020-04-20T11:05:00</start>
    <end>2020-04-20T15:50:00</end>
    <operator>Hans Wurst</operator>
    <purpose>Zeit totschiagen</purpose>
  </general>
  <sample>
    <name>Zufallsprobe 1</name>
    <description>Vom Nachbarn gemopst</description>
  </sample>
  <comment>Sein oder nicht sein...
</coment>
</root>
```

Listing 4: Original-Infodatei

common Info file - v. 0.1.0

GENERAL

Date start: 2020-04-04
Time start: 11:05:00
Date end: 2020-04-04
Time end: 15:50:00
Operator: Hans Wurst
Purpose: Zeit totschiagen

SAMPLE

Name: Zufallsprobe 1
Description: Vom Nachbarn gemopst

COMMENT

Sein oder nicht sein...

Listing 5: Infodatei im JSON-Format

```
{
  "format": {
    "type": "common Info file",
    "version": "0.1.0"
  },
  "general": {
    "start": "2020-04-04T11:05:00.000Z",
    "end": "2020-04-04T15:50:00.000Z",
    "operator": "Hans Wurst",
    "purpose": "Zeit totschiagen"
  },
  "sample": {
    "name": "Zufallsprobe 1",
    "description": "Vom Nachbarn gemopst"
  },
  "comment": "Sein oder nicht sein...\n"
}
```

Listing 6: Original-Infodatei

```
common Info file - v. 0.1.0

GENERAL
Date start: 2020-04-04
Time start: 11:05:00
Date end: 2020-04-04
Time end: 15:50:00
Operator: Hans Wurst
Purpose: Zeit totschiagen

SAMPLE
Name: Zufallsprobe 1
Description: Vom Nachbarn gemopst

COMMENT
Sein oder nicht sein...
```

Listing 7: Infodatei im YAML-Format

```
---
format:
  type: common Info file
  version: 0.1.0

general:
  start:
    date: 2020-04-04
    time: 11:05:00
  end:
    date: 2020-04-04
    time: 15:50:00
  operator: Hans Wurst
  purpose: Zeit totschiagen

sample:
  name: Zufallsprobe 1
  description: Vom Nachbarn gemopst

comment: >
  Sein oder nicht sein...
```

Listing 8: Original-Infodatei

```
common Info file - v. 0.1.0

GENERAL
Date start: 2020-04-04
Time start: 11:05:00
Date end: 2020-04-04
Time end: 15:50:00
Operator: Hans Wurst
Purpose: Zeit totschiagen

SAMPLE
Name: Zufallsprobe 1
Description: Vom Nachbarn gemopst

COMMENT
Sein oder nicht sein...
```

- ▶ rein lokal: unabhängig vom Netzwerk
 - netzwerkbasierte Lösung scheitert ggf. an zu alten Messrechnern und am mangelnden Platz für einen weiteren Rechner im Labor.
 - Dokumentation direkt auf dem Messrechner: kein Medienbruch
- ▶ unabhängig von Messdatenformaten
 - Alternative Formate setzen mitunter voraus, dass man selbst das Format der Messdaten bestimmen kann.
 - Auch Messdatenformate, die Metadaten mitschreiben, sind keine echte Alternative, zumal die Metadaten meist nicht hinreichend flexibel sind und mitunter *falsche* Werte mitgeschrieben werden.
- ▶ sehr einfache Struktur des Formats
 - Infofile-Parser sehr einfach zu schreiben
 - Python-Referenzimplementierung ist quelloffen und dokumentiert



- ❏ Daten ohne Kontext (Metadaten) sind wertlos.
Die Information *während* der Datenerhebung ist maximal.
- ❏ Eine Formalisierung der Metadatenaufnahme ist essentiell für Vollständigkeit und reale Umsetzbarkeit.
- ❏ Metadaten sollten maschinenlesbar vorliegen, um bei der Datenverarbeitung genutzt zu werden.
- ❏ Ein Dateiformat für Metadaten muss menschen-schreibbar sein, ohne die Maschinenlesbarkeit zu korrumpieren.
- ❏ Kernaspekte sind Modularität und Erweiterbarkeit, wesentliche Felder Zweck und Kommentar.