
towards a
standardised file format
for
reduced reflectometry data

Jochen Stahn Paul Scherrer Institut, Switzerland

Andrew McCluskey European Spallation Source, Sweden

on behalf of ORSO

ORSO

about us

we are a loosely organised group of scientists
somehow involved in **x-ray** or **neutron reflectometry**

with the aim to improve this technique through collaboration

we do so in 4 working groups:

education & outreach



file formats

data analysis



reproducibility

please contribute with your expertise and experience!

contact reflectometry.org





file formats

working group

ANSTO Andrew Nelson

diamond Tim Snow

ESS Andrew McCluskey, Tom Arnold

ILL Nina Steinecke, Thomas Saerbeck

ISIS Andrew Caruana, Arwel Hughes, Christy Kinane, Jos Cooper
[Max Skoda](#), Rob Dalglish

JCNS Joachim Wuttke

NIST Alexander Grutter, Brian Maranville

PSI Artur Glavic, [Jochen Stahn](#)

Univ. Kiel Bridget Murphy

Univ. Uppsala Adrian Rennie

and many more

mostly european
mostly neutron experts
mostly beamline scientists
mostly male
→ please help to balance!





file formats

motivation

```
# eos reducer script for AMOR Selene guide measurements output file
```

```
# [info] what was measured?
```

```
# call = "eos -a 0.02 -y 1,28 -q 0.01,0.04 -l 3.5,14. -F 0.01,0.012 -SRlt -r 12680 -n 12645"
```

```
# time = "2021-12-06T17:11:27"
```

```
# eos_version = "1.0"
```

```
# system = "amor.psi.ch"
```

```
#
```

```
# [parameters]
```

```
# omega = 0.699 # deg
```

```
# monitor = 1.0 # micro A s (proton charge)
```

```
# theta_range = [0.349000000000000053, 1.04900000000000004] # deg
```

```
# lambda_range = [3.5, 14.0] # Aa
```

```
# temperatur =655.2 # K
```

```
#
```

```
# [input files]
```

```
# references = [
```

```
#   "../raw/amor2021n012680.hdf", # 2021-12-06T17:11:27
```

```
#   ]
```

```
# datafiles = [
```

```
#   "../raw/amor2021n012645.hdf", # 2021-12-06T17:11:27
```

```
#   ]
```

```
#
```

```
# [data]
```

```
#   q_z      | R(q_z)      | Delta q_z    | sigma R(q_z)
1.01010101e-02  1.31834472e+00  1.01010101e-04  1.59818770e-01
1.03050709e-02  9.57938640e-01  1.03050709e-04  1.22878462e-01
```

units?

e.g. former reflectivity file at PSI



file formats

principles vs. **pragmatism**

- inter-operability** data to be processed by a wide **variety** of software
- reusability** sufficient information for further processing or **interpretation**
- correctness** **quantities** are well defined, labeled and have a unit
- ownership** of the original and the processed data
- reproducibility** information to recreate from the raw data
- practical aspects** limited availability of information
how to include information
- acceptance** old **habits**
established use of terms
- usability** **human** and computer readable
clearly laid out
- open for** individual demands
future developments



file formats

projects

dictionary definition of keywords

rules about units ...

definitions of terms and quantities

representations pragmatic **ASCII** file .ort
→ readability

comprehensive **HDF5** file .orb
→ complex data sets
→ future analysis concepts

tools python modules for writing and reading orsopy

ASCII representation requirements

clientele **users** → compatibility & extended information
beamline scientists → data policy, information flow
programmers → standardised I/O

aim **easily human readable**

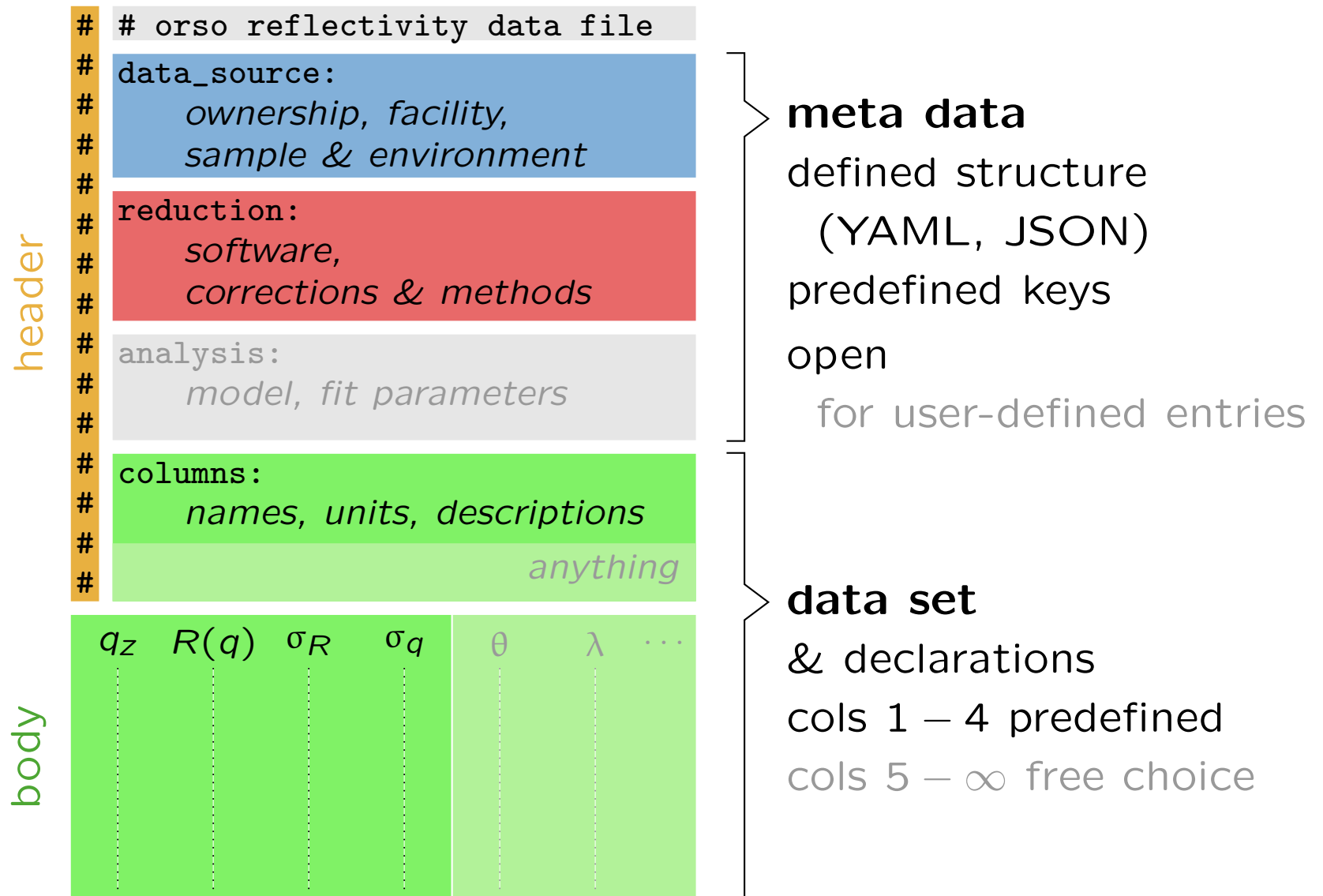
header with a defined **minimum** of meta data
extended set of **optional** entries
rules for extra entries
computer readable

body with a defined structure
predefined columns
accepting any number of additional columns
capable of containing several sets of data

reflectometry.org/file_formats



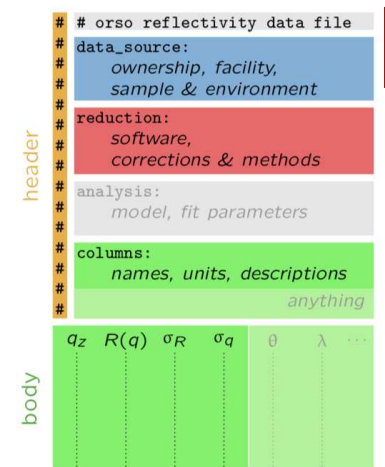
ASCII representation structure



ASCII representation

demo

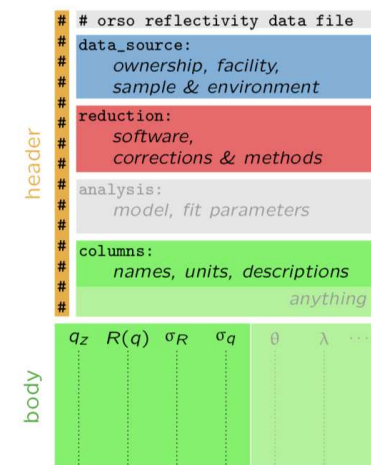
```
# # ORSO reflectivity data file | 0.1 standard | YAML encoding | https://www.reflectometry.org/
# # Interdiffusion in Fe | 2020-12-24 | sample fe-457-2 | time resolved | T = 800 K
# data_source:
#   owner:
#     name: Jochen Stahn
#     affiliation: PSI, CH 5232 Villigen
#     contact: jochen.stahn@psi.ch
#   experiment:
#     title: Interdiffusion in Fe
#     probe: neutrons
#     facility: PSI SINQ
#     instrument: Amor
#     proposalID: 2021 9876
#     start_date: 2021-05-16
#   sample:
#     name: fe-457-2
#     composition: [ Fe (80 angstrom) | 54Fe (20 angstrom ]_12 | Si
#     description: 10 x 10 mm^2
#     environment: small in-situ furnace with improvised permanent magnetic field
```



ASCII representation

demo

```
# reduction:
#   software:
#     name:      eos
#     version:   1.2
#   call:       eos -n 1234 -r 1111 -e -SRlt FeFe1
#   timestamp:  2021-09-22T12:45:15
#   creator:
#     name:      Jochen Stahn
#   corrections:
#     - footprint
#     - background
#     - ballistic correction
#     - incident intensity
#     - detector efficiency
#     - scaling / normalisation
```



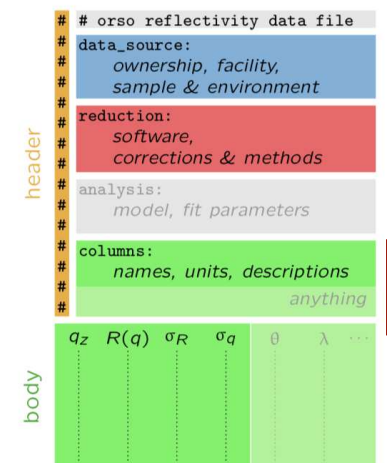
ASCII representation

demo

```
# columns:
#   - name:      Qz
#   unit:        1/angstrom
#   description: wavevector transfer
#   - name:      R
#   description: reflectivity
#   - name:      sR
#   description: standard deviation of reflectivity
#   - name:      sQz
#   unit:        1/angstrom
#   description: standard deviation of wavevector transfer resolution
#   - name:      alpha_i
#   unit:        deg
#   description: angle of incidence
# data_set:  0
```

#	#	Qz	R	sR	sQz	alpha_i
1.03563296e-02	3.88100068e+00	4.33909068e+00	5.17816478e-05	1.000000000e-1		
1.06717294e-02	1.16430510e+01	8.89252718e+00	5.33586471e-05	1.100000000e-1		

...



orsopy

in a nutshell

python modules to **read and write** .ort files
as interface to SLD data base

authors A. Glavic, B. Maranville, A. McCluskey, A. Nelson

implemented in
analysis

refnx
GenX
Refl1d
easyReflectometry

reduction

eos (PSI)
scipp (ESS)
reductus (NCNR)
POLREF (ISIS)

info & docs <https://github.com/reflectivity/orsopy>



orsopy

implementation

installation `> pip install orsopy`

usage

writing

```
import numpy as np
from orsopy.fileio import Orso, OrsoDataset,
    save_data, Person
metadata = Orso.empty()
# populate metadata
metadata.data_source.owner = Person('J. Stahn',
    'PSI, CH 5232 Villigen', 'jochen.stahn@psi.ch')
...
data = np.array([q, R, qR, dq]).T
save_data(OrsoDataset(metadata, data), 'data.ort')
```

reading

```
from orsopy.fileio import load_data
metadata_rich_data = load_data('data.ort')
```

info <https://orsopy.readthedocs.io>



orsopy

extension: analysis results

```
# analysis:
#   software:
#     name: GenX
#     version: 3.5.6
#   script: "import models.spec_nx as model\nfrom models.utils import UserVars, fp,\
#     \ fw, bc, bw\nfrom numpy import *\n\n# BEGIN Instrument DO NOT CHANGE\nfrom models.utils\
#     \ import create_fp, create_fw\ninst = model.Instrument(probe='neutron', wavelength=4.4,\
#   ...
#   parameters:
#     - Parameter: Si0.setD
#       Value: 1211.2966080978158
#       Fit: true
#       Min: 903.75
#       Max: 1506.25
#       Error: '-'
#   ...
#   statistics_mcmc:
#     library: bumps
#     version: 0.8.0
#   ...
#   operator:
#     name: Artur
#   timestamp: '2021-12-09T17:12:19'
```

.ort output from GenX
including model and fit parameters

- structure by ORSO
- keys and content by Artur
- orsopy extension by Artur

ORSO

is essentially **OPEN**

contribution feel free to **join, comment and criticize!**

content small set of default entries

lots of optional entries provided

expandable you can add (almost) any **content** you like

contact reflectometry.org

THANKS to everyone who contributed
to you for listening

