the file formats working group

is ready to release a

standardised file format

for

reduced reflectometry data

Jochen Stahn Paul Scherrer Institut, Switzerland

on behalf of ORSO

the file formats working group

consists of / got suppot from

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 Dalgliesh
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

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 clear lay 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 → readability	`.ort`
	comprehensive **HDF5** file → complex data sets → future analysis concepts	`.orb`
tools	python modules for writing and reading	`orsopy`

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

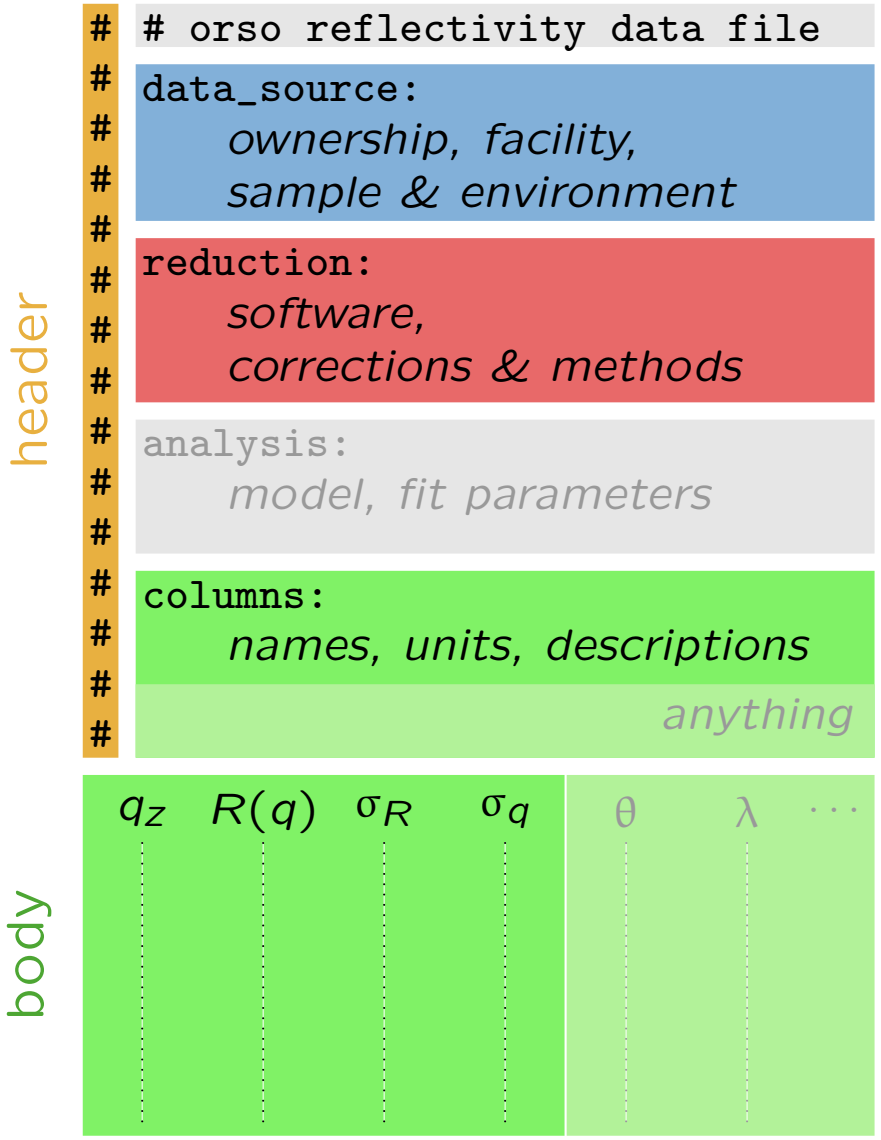
aim **easily human readable**

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

ASCII representation

structure



meta data

defined structure
(YAML, JSON)

predefined keys

open
for user-defined entries

data set

& declarations

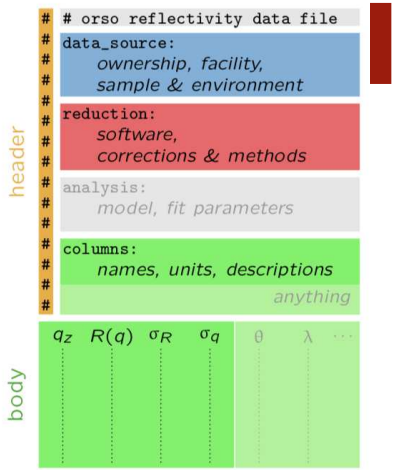
cols $1-4$ predefined

cols $5-\infty$ free choice

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:          neutron
#         facility:       PSI SINQ
#         instrument:     Amor
#         proposalID:     2021 9876
#         start_date:     2021-05-16
#     sample:
#         name:           fe-457-2
#         description:    10 x 10 mm^2
#         environment:    small in-situ furnace with improvised permanent magnetic field
```

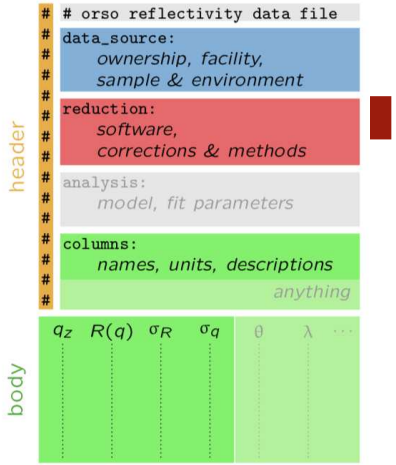


reflectometry.org/file_formats

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
```

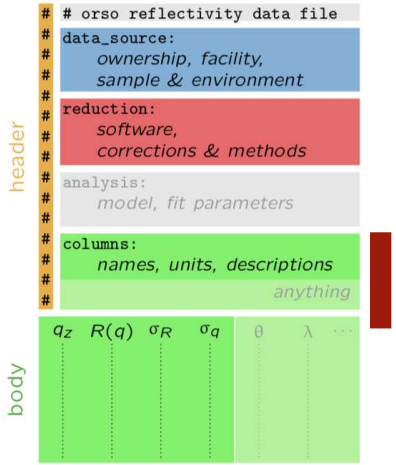


reflectometry.org/file_formats

ASCII representation

demo

```
# columns:
#       - name:          Qz
#         unit:          1/angstrom
#         description:   wavevector transfer
#       - name:          R
#         description:   reflectivity
#       - error_of:      R
#         error_type:    uncertainty
#         value_is:      sigma
#         distribution:  gaussian
#       - error_of:      Qz
#         error_type:    resolution
#         value_is:      FWHM
#         distribution:  rectangular
#       - 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
...
```

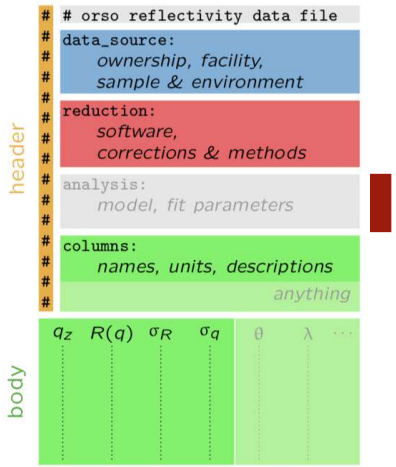


reflectometry.org/file_formats

ASCII representation

demo - optional content

```
# 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\nf
#     \ import create_fp, create_fw\ninst = model.Instrument(probe='neutron', 
...
#   parameters:
#   - Parameter: SiO.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 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://orsopy.readthedocs.io

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, sR, sQ]).T
save_data(OrsoDataset(metadata, data), 'data.ort')
```

reading

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

file formats

on release of .ort specs and orsopy

Do you accept the specifications for the ASCII representation of the ORSO file format for reduced reflectivity data?

If approved, `orsopy 1.0` will be released.

please

have a look at the documentation

make up your mind

vote

← specs

orsopy →

file formats

conclusion

content small set of default entries

lots of optional entries provided

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

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

THANKS **to everyone who contributed**

to you for listening

please

← **have a look at the documentation** →

make up your mind

vote

specs

orsopy