

# EQSANS Data Reduction with drtsans – Quick Reference for Jupyter Notebook Mode

Editors: Changwoo Do, Gergely Nagy, Yingrui Shang, William Heller  
Date: 07/26/2021

## Introduction

EQ-SANS data reduction is typically performed through three equivalent means:

- via scripts run in the terminal window,
- via mantidworkbench or
- via jupyter notebooks

The present document details the third method. Your local contact will work with you to configure the reduction for the various instrument configurations that you use during your experiment.

Below please find the step-by-step description of the procedure.

## 1. Visit [jupyter.sns.gov](https://jupyter.sns.gov)

- Data reduction can be performed both from inside and outside of ORNL
- Log in with your xcams ID and password
  - The same credentials are needed to transfer files to your personal computer via e.g. WinSCP

## 2. Find the EQ-SANS example reduction scripts

- Click on the examples menu option



- Depending on the number of settings used in your experiment use (click the green 'Use' button next to the file name)
  - eqsans\_reduction\_example\_1config.ipynb
  - eqsans\_reduction\_example\_2config.ipynb
  - eqsans\_reduction\_example\_3config.ipynb
- After having created your own copy of the example file in the kernel menu change kernel to \*[analysis-cluster] sans

## 3. Modify the reduction file to fit your experimental protocol

- Update output directory

### USER INPUT ¶

```
In [13]: # define the IPTS number and the output directory
         ipt_s_number      = 26695
         output_directory  = "/SNS/snfs1/instruments/EQSANS/IPTS-26695/shared/example/"
```

- It is recommended that the target directory for the reduced files be empty or to not to contain the results from previous reductions in EQSANS\_runnumber\_\*.nxs format because drtsans will over-write previous results without asking the user if they wish to do so.
- Update the run numbers and names for individual scattering and transmission runs, which in this example correspond to the standard measurements of porous silica

```

#define the sample names and thicknesses
sample_names = ['porsil']
sample_thick = [1] * len(sample_names)

#define the scattering and transmission data for the lowest Q setting
samscatt_conf_0 = ["123162,123209"]
samtrans_conf_0 = [123157]
bkgscatt_conf_0 = ["123161,123208,123222"] * len(sample_names)
bkgtrans_conf_0 = [123156] * len(sample_names)
emptybeam_conf_0 = 123155

```

- c. Add run numbers and names for individual scattering and transmission runs for the samples
  - i. Multiple samples can be inserted with comma separation, e.g. 123456, 123457
  - ii. Multiple runs collected from a single sample that you wish to sum together are specified as a comma-separated list enclosed in quotation marks, e.g. "123456,123457"
  - iii. One can also use '-' to list a consecutive list of samples. For example, [123612-123712] is a list of 101 samples
- d. To look up run numbers from your experiment, please refer to
  - i. oncat.ornl.gov
  - ii. or the catalogue method presented in the other tutorial document.
- e. Comment all unnecessary sections of the file by using either a # to remove a single line or a pair of ''' at the start and end (i.e. enclosing) of sections that you wish to comment out and save the document
  - i. This is advised to prevent time consuming re-running of already completed reduction runs
- f. If you are using a 2 or 3 configuration template file update the overlap array. While the examples present stitching of two or three data sets, it is possible to stitch data from several different configurations together. Then, the overlap array takes the form "overlap = [MergeAB\_min, MergeAB\_max, MergeBC\_min, MergeBC\_max, ...]", having (Num\_Configurations -1) pairs of minimum and maximum Q-values for the merge regions.

```

# stitching overlaps
stitching_q_0_1_2 = [0.02,0.03,0.07,0.08]
target_index = 0

```

- g. The stitching ranges are generally determined by examining the data files. Feel free to ask your local contact or another instrument scientist for advice.
- h. The stitching ranges must be listed in order of increasing momentum transfer (from low-Q to high-Q)
- i. During scaling of the independent curves target\_profile\_index will determine the curve to scale to. **Note that index of the first curve is 0.**
- j. Save the file
- k. Run the file cell by cell or running the entire document
  - i. Depending on the size of the data files, which can be quite large for strongly scattering samples, each data set may require several minutes to reduce because the instrument saves data in "event mode", which results in large files. If you are reducing a large number of files, such as after an overnight script, you may want to get a cup of coffee.
- l. Reduced files can be found in the output folder in a variety of formats.
  - i. The data review section of the templates shows examples for displaying the 1 and 2D datasets.

#### 4. Data reduction with added flexibility

- a. The template file provided serves as a reduction example for simple cases. The data reduction software is highly configurable through the various reduction parameters that can be seen in the json files created during script execution. Consult with an instrument scientist about their usage. Data reduction with more specialized needs can be achieved through modifying json parameters. A separate document describes the various parameters.
- b. The json files can be directly edited with gedit or pluma, but can be viewed easily with the prettyjson.sh script that is created during the steps described in section 3.
  - i. Being in the output subdirectory within the shared directory the script can be run as follows:
    1. ./prettyjson.sh <name\_of\_jsonfile>
- c. Modify the selected json parameters by adding new lines to the reduction script. In the example below, custom time-of-flight values are defined for limiting the range of neutrons of interest.

```
cuttofmin_conf_0 = 4000
cuttofmax_conf_0 = 4000
```

- d. Other parameters can be modified in the staff input and execution section

```
eq0 = EQVar('2020B')
eq0._outputdir = output_directory
eq0._ipts = ipts_number
eq0._standardabsolutescale = 27.9834
eq0._numqbins = 100
eq0._qbintype = "log"
eq0._showjson = True
eq0._thickness = sample_thick[i]
eq0._bkgscatt = str(bkgscatt_conf_0[i])
eq0._bkgtrans = str(bkgtrans_conf_0[i])
eq0._samscatt = str(samscatt_conf_0[i])
eq0._samtrans = str(samtrans_conf_0[i])
eq0._empty = emptybeam_conf_0
eq0._filename = str(sample_names[i]) + "_conf_0"
eq0._cuttofmin = cuttofmin_conf_0
eq0._cuttofmax = cuttofmax_conf_0
eq0._maskfilename = maskfile_conf_0
eq0._sensitivityfilename = "/SNS/EQSANS/shared/NeXusFiles/EQSANS/2021A_mp/Sensitivity_patched_thinPMMA_4m_120578.nxs"
reduceJupy(eq0)
```

- e. All parameter names are to be written in lowercase letters.

#### 5. Troubleshooting

- a. It is highly recommended to clear the results every time one reruns the results from Menu->Kernel->Restart & Clear Output
- b. Having too many jobs hanging and unterminated may result in a "Dead kernel" problem. Please make sure all unnecessary jobs are closed from the Running tab (next to the Example tab in Jupyter home page), and try again

#### Disclaimer

The data reduction scripting method presented here was developed by Dr. Changwoo Do ([doc1@ornl.gov](mailto:doc1@ornl.gov)), and the jupyter examples are maintained by Dr. Gergely Nagy ([nagygy@ornl.gov](mailto:nagygy@ornl.gov)). Responsibility for its use, bug fixes and further development lie solely with the EQ-SANS team. All questions and concerns should be directed to your local contact, who will either help debug your reduction script or provide feedback to the Research Software Engineering group about bugs in the drtsans package that performs the reduction.