Notes from LSS/CS discussion on reflectometry.

March 17 2015 @ 14 hrs

Present: Bob C, Yuri G, Thomas, S, Philipp G, Richard C, Miguel G, Eric P, Didier R, Mark J

Objective: requirement capture and description of work for current and future reflectometry techniques to be implemented in Mantid.

Notes from meeting/discussion and subsequent thoughts

Methods and algorithms

- Methods have been described separately as linear workflows
- Algorithms (e.g. peak/background determination) have been identified as those operations that may not be available as standard operations in the Mantid framework (e.g. addition, multiplication, etc of workspaces).
- Algorithms must however be generic, as far as possible, so that they can be widely used.
- The 'devil is in the detail' - methods and algorithms must be defined in sufficient detail, indicating corresponding COSMOS/LAMP algorithms where relevant.
- A coherent set of methods (e.g. TOF-based) can be combined in an application based on a flow chart which shows the common and distinct parts of the data treatment and defines the objects and methods of the application (the 'orange' flowchart is a very good example with an appropriate level of detail)

Quality control

- Easiest to base validity of new software on standard data sets and benchmarking against existing codes. Collect as many standard data sets as possible.

Mantid

- Check which ISIS (SNS?) instruments use multi-detectors and are therefore similar to D17 and FIGARO
- Check if there is already a mechanism for tracking changes to a subset of Mantid algorithms e.g. for reflectometry - the basic information is certainly available since every code modification is recorded on the software forge (GITHUB) and/or via the Jenkins build server.
- There are 2 official releases per year. The code is 'built' every night and made available for download. Only automatic tests are covered by the nightly build. Full (human) testing e.g. of interfaces is performed for the official releases. The code version with the appropriate functionality must be used (YG).
- Mantid is a framework and library. The interface is called MantidPlot and it is completely separate in terms of code from Mantid (BC).
- ILL reflectometry in Mantid should/must use as much of the existing algorithms and interfaces as possible – the decision on what can be used will be taken at the start of the Mantid-COSMOS project (PG).

COSMOS/GALAXY

- Mantid reflectometry GUI is currently less ergonomic than COSMOS (YG)

- COSMOS is not currently used for monochromatic measurements (theta-2theta scans) (TS)
- Performance issue with COSMOS – it becomes slow when handling a very large number of data sets (> 1000)

Methods

- theta-2theta scans are used all the time on SuperAdam, about 15% of the time on D17 and never on FIGARO. This data is treated with scripts in LAMP, a prox file/sheet can be used on SuperAdam, but they also have their own set of programs (based on Igor?)
- Q_X-Q_Y-TOF data is read into LAMP, manipulated/sliced and output as simpler e.g. 1D data sets.
- Rainbows-type data has not been discussed
- Coherent sum method should be described as a separate method
- Event mode data - required to monitor fast, cyclic processes which are not perfectly periodic. As for SANS, it is proposed to trial a time binning scheme and treat like kinetic data with variable time bin widths. Then repeat/optimise time binning to give suitable changes and statistics in reduced data. Note that Mantid handles event mode data natively since all SNS data is acquired in this mode.
- Other methods discussed are kinetic data and polarised neutrons
- All methods will be based on a set of operations and algorithms which are generic (i.e. not specific to reflectometry data) as far as possible and be callable as scripts.
- All methods or a coherent subset (e.g. all those involving TOF data) will be accessible through a GUI.
- A large number (> 200 ?) of algorithms/routines have been written in LAMP/IDL, many are obsolete - the useful ones need to be identified.

NeXus files

- There is no limit to the complexity of data that can be stored in this type of data file.
- The tradition in reflectometry is to store a data set corresponding to one instrument configuration in one data file.
- Storing many data frames in one file makes sense (reduced number of files, easier/more efficient access to data in files,...) when one instrument variable changes: time => kinetic data, polarisation states =>, detector positions => theta-2theta scans
- Note: on D7, the different polarisation data for one detector position are stored in the same NeXus file. When a second detector position is required to complete the detector coverage, this data is in a second, separate file.

Interaction with NoMad

- A prototype, live data analysis server exists which allows NoMad to call any 'callable' data reduction and analysis software e.g. LAMP or Mantid with a script.
- At a 'basic level', all methods therefore have to be implemented as callable scripts.
- NoMad, in the future, can be the point at which spreadsheets of measurements (and the corresponding data) are defined and then used to call the data treatment directly or saved for use with data treatment independently i.e. automatic filling of tables for COSMOS and/or GALAXY.