Martin Dove's RMC Workflow Diagram

Erica Yang

e-Science centre
Rutherford Appleton Laboratory
Science and Technology Facilities Council
(erica.yang@stfc.ac.uk)

July 7, 2010

1 Background

The RMC diagram, depicted in Figure 1, describes the *data analysis workflow* for one of ISIS's instruments. It was produced by Prof. Martin Dove for the I2S2 project. This report is based on that diagram. The diagram has been redrawn using Microsoft Visio software and is shown in Figure 2. Most of the technical explanation is extracted (in most cases, this means copied) from Martin's emails and my interactions with him and his team during my visit to ISIS (hosted by Prof. Martin Dove and Dr. Matt Tucker) on 14th Dec. 2009.

2 The Workflow

My definition of a workflow is an ordered (linear and/or parallel) sequence of processes driven by human (scientists and/or instrument scientists) behaviours.

Such a workflow often involves

1. **Programs**: a program is a state machine which takes one or more inputs (files or human manual inputs) and generates outputs, such as files or feedbacks (e.g. as a plot or a file) to human as intermediate outputs.
2. **Inputs and outputs**: inputs to and outputs from programs
   (a) Files: typically the workflow generates many files, and these may include many of the same type as well as many different types.
   (b) Human manual inputs/guidance: this is a very important aspect to be included in the picture because they represent human knowledge/experience and follow from the analysis, which is essential to the experiment (e.g. for others to understand the results, reproduce/validate the analysis/outputs, continue the work from where the last scientist left).
   (c) Human generated files: experiment- and run- specific information plus analysis oriented (e.g. depending on what atom scientists want to analyse) parameters
   (d) Program feedbacks: these are intermediate feedbacks from programs to scientists, allowing to determine how to steer the analysis and whether satisfactory results have been obtained. Although they are 'intermediate' (not stored, therefore no need to make available), they are included in the diagram to show human activities are present in the workflow and typically it is a repeated process (until the scientist is happy to proceed). Because of the workflow involves extensive human judgement, it can take months to finish the analysis.

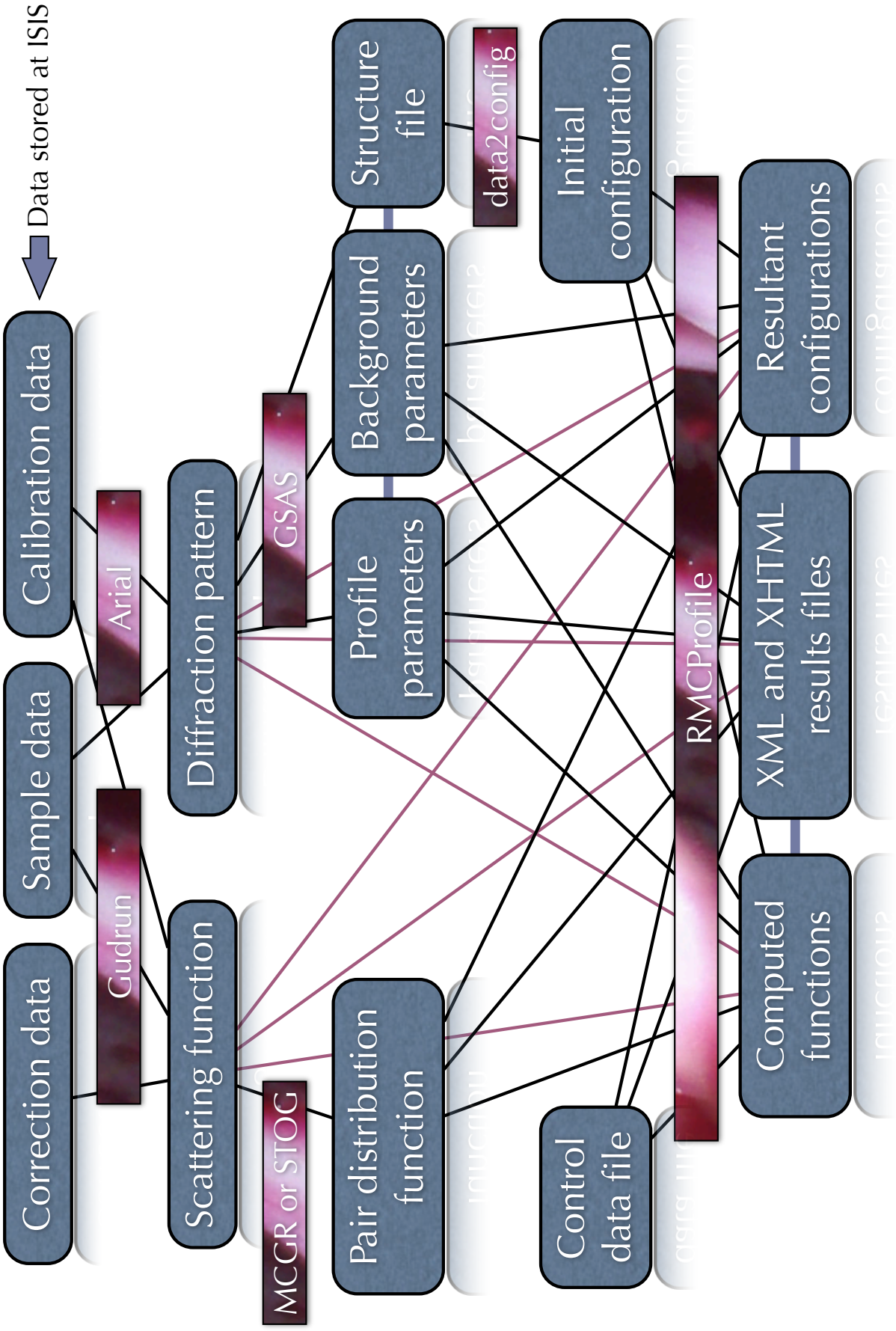

Figure 1: Martin Dove's Original RMC data flow diagram

3. **Information flow**: the connections between inputs and outputs, and programs.

The files generated from this work are grouped into three categories:

1. raw data: instrument specific, collected by detectors, and set by instrument scientists
2. derived data: there could be many files containing such data, these are generated in parallel by different programs
3. resultant data: data fit for being presented in a paper/talk etc.

Figure 2 shows the workflow of analysing the experimental data generated from one of the ISIS's instruments, called GEM, using a method called Reversed Monte Carlo (RMC). Scientists' workflow varies depending on a number of factors:

1. type of facility,
2. type of instrument,
3. goal of the experiment,
4. analysis methods (e.g. RMC - Reversed Monte Carlo), and
5. availability of analysis programs (most programs are specifically written for a specific instrument, sometimes it is only available in a particular facility, although some are open source/access).

For example, the workflow for analysing the same set of experimental data will be different if a different analysis method is chosen.

3 Programs

Except Ariel (in IDL), all the programs are written in Fortran.

Gudrun is a difficult procedure[1] that the scientist runs, and it generates what we call the Scattering function. We produce one of these functions for each bank of detectors. This is an ISIS-specific data reduction tool [2]. Gudrun has, but is the only one in this workflow, a Java GUI.

In this process, human has to worry about the role of noise in the data in going from scattering function to pair distribution function. Noise can arise from several sources, e.g. errors in the Gudrun data reduction, or noise due to the statistics on the data. The way that one applies MCGR or STOG may need some care.

Ariel an IDL program [3] which corrects the data to be ready for a diffraction analysis, and this is an automatic process (again, there will be separate data for different banks of detectors). Again, this is an ISIS-specific data reduction tool.

MCGR or STOG performs the Fourier transform of the Scattering function to produce the Pair distribution function (PDF). The PDF is simply a histogram of all distances between atoms. (I would remark at this point that) the Scattering function and PDF have some absolute values we must take note of, including values and certain integrals. If these don't come out consistently, then we iterate around the procedure until they do.

[1] For data that are free of problems, this procedure should not be difficult at all. The difficulties arise particularly when we have hydrogen in the sample, and in this case some of the data correction equations are outside their range of applicability.

[2] Gudrun was written in Fortran with a Java GUI by a couple of scientists at ISIS (Dr. Alan Soper), and they give it away in executable form from http://www.isis.stfc.ac.uk/instruments/sandals/data-analysis/gudrun8864.html

[3] IDL is now out of fashion and, like Matlab, is expensive for people to buy. Moreover, because products such as IDL can change their API's without care for the user, tools such as Ariel will be tied to a specific version of IDL, which is not nice nor sustainable. Thus the decision has been made to no longer maintain Ariel but to move away from it.

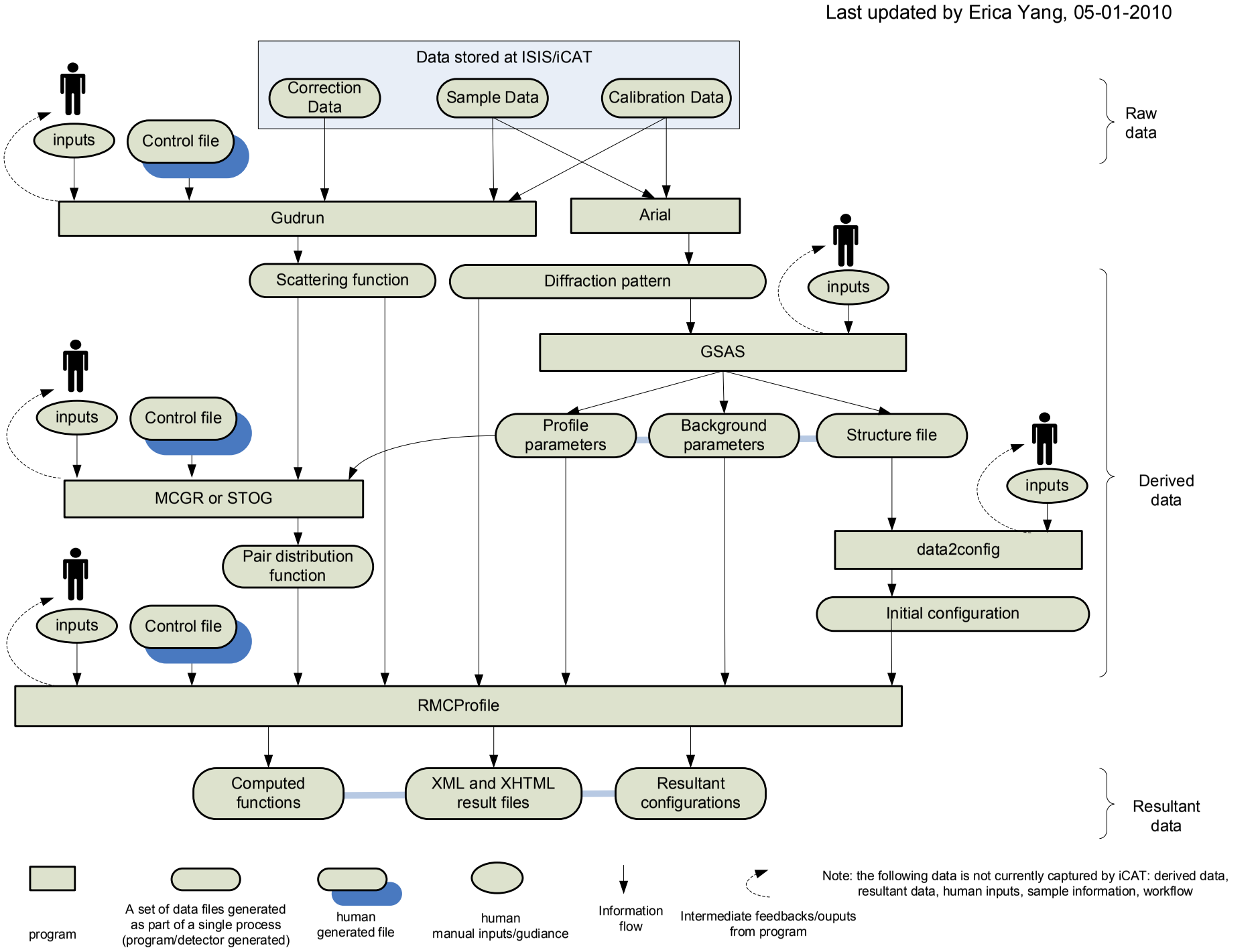

Figure 2: Martin Dove's RMC data flow diagram (MS Visio)

GSAS performs a Rietveld analysis using the GSAS code (this is one stage that Simon does with his Diamond data [4]). Put simply, the Structure file gives information about the average positions of the atoms within the repeating unit in the crystal and the amplitudes of their fluctuations. If you build a histogram of distances between the average positions, you will find that it is not completely consistent with the PDF histogram. This is because the PDF contains information about the correlated fluctuations of the positions of the atoms. Put together, the PDF and structure tells you something about the long-range order and short-range fluctuations of the atoms. As part of GSAS you obtain information about the parameters describing the resolution of the diffraction experiment, and you also obtain parameters that describe the background in the diffraction pattern.

The human has to control the GSAS side too, to the same degree. GSAS is the approach to fitting the crystal structure to the diffraction pattern, and like all regression methods it needs to be nursed or else it can go wrong (since regression methods follow a line in the multi-dimensional phase space in a determinate way, and this line may not lead directly to the true minimum).

GSAS is a closed-source but free program, that is distributed in versions for the major operating systems. In that sense it can be said to be proprietary. It does have a GUI[5] as well as being driven by command line.

data2config Then the human has to decide what size configuration you need, which comes in the data2config stage. data2config is indeed a simple[6]tool (a mere 3600 lines of code) that will generate configurations from a range of input files, such as those generated by GSAS, or available from crystal structure databases for example. It is open source and free.

RMCProfile is the key part of the analysis. This uses all the derived data produced so far, with one caveat, namely that the Structure file is used to generate an initial configuration[7]of atoms that is to be used and which consists of some multiple (say $8 \times 8 \times 8$) of the repeating unit represented by the Structure file. We also need a Control data file to drive the RMCProfile code. This analysis is likely to take **several days** of computer time.

RMCProfile is a very human-oriented activity. RMCProfile is a code we[8] have written, and is a significant analysis code.

4 Files/inputs/outputs

The data are in the domain of crystallography. The diffraction part is mainstream crystallography. The Scattering factor side is more specialised but this is a growing area.

Sample data represents the measurements on 5000 detectors grouped into around eight banks.

Calibration data consists of a number of files that are established once in a while, which typically determine the angle the neutron beam is scattered through when going from the sample to the detector.

[4]GSAS is an example of a code that performs Rietveld analysis, which is a way of tuning a crystal structure against the experimental powder diffraction data (neutron or x-ray) using a least-squares method. For the crystallographer, the key variables are the unit cell parameters (i.e. the repeat distances in the crystal), the atomic coordinates within a single repeating unit (the unit cell), and if possible the amplitude of fluctuations of the positions of the atoms. Other things that get tuned are the background function, parameters that control the shapes of the diffraction peaks, and possibly angle offset errors. Many experiments at ISIS and Diamond will use the Rietveld method to refine crystal structures.

[5] GSAS has an interface (via X11 and TCL on mac).

[6] (But in fact, it is) not really simple; for example, CIF files are very hard to program against because the specifications are rather too flexible.

[7] The initial configuration is what data2config produces. I am not sure why there is a caveat.

[8] http://wwwisis2.isis.rl.ac.uk/rmc/

Correction data [9]corresponds to our own measurements performed at the time we collect the Sample data, and consists of measurements of the empty instrument, an empty furnace or cryostat, and an empty can, together with a spectrum from a rod of vanadium to ensure that all detectors are normalised against a common absolute standard.

Computed functions there are many of such and they include calculated versions of the data together with all the intermediate functions that are required to generate them.

XML/XHTML There is a main output text file, but we like to work with XML output files that are transformed to XHTML files that highlight the information content of the data.

Configurations of atoms RMCProfile also produces new configurations of atoms, which are perhaps the main outputs of this method because they can then be analysed further.

5 Other useful stuff

5.1 Diamond experiments

1. 100G per experiment (3 days), 1 G per 2 hour experiment
2. all the raw data files generated at Diamond can be processed by scientists on their own laptop because the raw data is generated on the same type of instrument as the ones they use in their own lab. Diamond does not provide facilities to keep the derived data. But data from Diamond experiments is significantly substantial (in terms of file size) than the data from ISIS experiments.

5.2 ISIS experiments

This section describes some information about ISIS experiments.

5.2.1 Data and workflow

1. 30M per raw data file
2. roughly 100 raw data files per 3/4 day experiment (all of them are good files. there are no so-called bad raw files produced) every run takes between 6 to 8 hours. Runs are performed in ISIS all day long (i.e. 24 hours). There are 8 banks of detectors, with around 5000 detectors in total. So each bank contains many detectors, which are grouped together in the analysis. Each run is often broken down into smaller chunks in case something goes wrong. For example, if there is a temperature failure half-way through, you would lose all the data if you collected in one chunk. But if you collect in small chunks, the chunks before the failure may still be usable.
3. ILL vs. ISIS: Each facility develops the instruments they want, some of which complement and some of which compete. The workflow will not be the same for different instruments.

5.2.2 Gudrun and Scientists

Erica There is no meta-dependency between files (raw files, derived data files). Only instrument scientists know how to decipher/sequence the derived data files. For example, in Gudrun, the input file (parameters set for examining the detectors in the readings presented by the Java GUI of Gudrun) is not stored alongside with the run files and the outputted scattering function files.

The linkage between the input and the scattering function is only kept on paper notebooks by instrument scientists and every scientists have their own way of keeping the input parameters.

[9]also the flight path.

Martin To define the term "instrument scientist", he/she is the ISIS employee designated to help the user run the experiment. In some cases the instrument scientist may help get going and then leave the scientist to themselves. In our case the instrument scientist is part of our team.

It isn't actually the case that only the instrument scientist knows how to decipher or sequence the derived data files. What may be true is that if there is a new bit of ISIS software the instrument scientist will be the first person to know how to use it. This is the case with Gudrun, which has a new GUI that is being learned by the team.

To interpret the bit about your Gudrun example, there is a set of files that concern instrument parameters which are stored as files, and these are stored with the run files. There is also a Gudrun input file that is generated by the GUI (Gudrun itself runs as a standard executable with input and output as a separate program), and in the past we used to only use this file. Typically in our team Dave has taken on the role of doing the Gudrun work, and he stores the input file with the data, but only passes onto us the resultant scattering function and pair distribution function file.

What is *not captured* in any systematic sense is information about the sample, such as its density and the size of the sample can.

5.2.3 Paper notebook

Erica The current practice: instrument scientists put these parameters on their paper notebook (or save the input file on their own computer - i.e. keep separated from the raw files and the metadata files currently saved in the ISIS directory/iCAT). Without such parameters, it is hard to understand the outputs from Gudrun (i.e. the scattering functions). Effectively, that means that even with the raw data files (the top three boxes), other people are quite difficult to reproduce the derived data (the scattering functions). The linkage between the input and the scattering function is only kept on paper notebooks by instrument scientists and every scientists have their own way of keeping the input parameters.

Martin Indeed, and this is the problem! It is not only instrument scientists, it is the owner of the data, who may or may not be the instrument scientist. I would add this is captured in the Gudrun input file, so if someone had the Gudrun input file they could recover these data.

5.2.4 On-site vs. off-site

Erica Only until recently, most of the analysis has to be done on site (in the ISIS cabins!) because some users are new to the equipment, some users are not knowledgeable enough to understand how to process the raw data using Gudrun (instrument scientists' experience and knowledge is required to set various parameters). Also, instrument scientists' knowledge of the calibration data is essential to the correct interpretation of the raw data/plots using Gudrun.

Martin The issue of on-site may evolve. If scientists don't know what to do, then they have to be helped, and on-site help is easier for the instrument scientist. However, we should not assume that this is the current or future state. Scientists who know what they are doing can now perform this stage back home.

The instrument scientists don't have any special knowledge of the calibration data. These data are contained in files that are available to the external scientist.

5.2.5 Programs

Erica Most of the programs are proprietary. Some are open source (e.g. RMCProfile). Some (e.g. Ariel) are very old without source and will be replaced by Mantid[10] in the upcoming years. Some (data2config) is just a simple tool, i.e. not doing much analysis. All the programs are written in Fortran, some (like Gudrun) have Java GUI. Some (RMCProfile) are very advanced and powerful (produce plots, do transformations, XML/XHTML, annotation/dictionary for terminologies).

[10] `http://www.mantidproject.org/`

5.3 Motivations for preserving derived data

Erica Why derived data (e.g. the input + scattering function) needs to be preserved?

- others may want to validate the experimental results by reproducing the analysis.
- authors want to illustrate the steps how they come to the conclusion (resultant data) with the raw data. This makes their research more credible.
- by having the scattering function (and the input parameters), others can further analyse the raw and derived data following different paths.

Martin This is the essential question to ask. I would also add

- so that we can write a paper from our analysis (remember that being busy scientists there is a good chance we may not write the paper immediately the analysis is completed)
- so that we can use the data in other works, e.g. review papers, talks etc, again after coming back to the data some time later
- so that work carried out by one team member can be used by another; a common example being a student does the analysis and the supervisor writes the paper, so the supervisor needs to have a good access to the data even after the student has left.

It is worth remarking that the derived data after the scattering function takes a LOT longer to obtain that the raw data takes to collect. Losing the derived data could lose *many months* of work.

5.4 Requirements

Erica My understanding of the requirements:

- all the contextual information regarding the inputs to the programs should be captured.
- all the derived data files need to be achieved
- the workflow between the stages needs to be captured
- any implementation should not change the existing workflow of the scientists: a non-intrusive approach for implementation should be sought. It is not realistic to expect the changes to be made to the software. Ideally, all the software we develop in I2S2 should be independent from these existing software (e.g. Gudrun, Ariel) that scientists use. Maybe, develop a shell-based approach like Tortoise SVN? 'Independent', to me, also means that even generating web services interfaces for them seems a bit too much.

Martin This sounds like a good understanding. I agree that we can't get into all the codes, although RMCProfile can be changed (and I am very happy for this). Shell-based approaches are good.

We also don't capture notes about the analysis. It is sometimes the case that RMCProfile is run from scratch several times, and the reasons why are never captured.

6 Acknowledgement

This report will not be possible without the timely help and continuous support from Prof. Martin Dove. Here, I would like to express my sincere thanks to him.