Meeting Minutes for ***Orsopy and a standard model language***

10:00-12:00 (UK), 11:00-13:00 (Europe), 05:00-07:00 (US East coast), 19:00-21:00 (Australia East coast):

This session will start a discussion around the model language. The session will aim to understand what is missing from the current definition of the the language, for example, can it be adapted to improve representaion of models for biological systems.

What can be done to improve uptake?

- *How do we ensure all facilities give users data in .ort format.*
- *What is the added value to users for using this format (The ORSO text format is now reasonably well defined and the orsopy tool is available, but can the documention be improved).*

Finally is there some merit in defining a standard format (or formats) for fit results. There is often a bottleneck after running analysis to get the fit, the SLD profile(s) and the confidence intervals etc out of the analysis software into an appropriate format for publication. The vision here is to add some kind of export functionality into the analysis software to export those things in formats suitable for common plotting software (Origin, Sigmaplot, Python (Plotly, Matplotlib etc), Matlab etc). This would be similar to the various image formats - not a single format, but a small collections of standard formats

---

Open discussion:

Jos Cooper – has met with a developer and discussed how best to interpret the model language. We want the text to be both machine readable and human readable, which means the language used can either be very simple or very complex. However, if you create a stack and make it readable to humans, it is not always readable by the computer.

Jochen Stahn – Artur Glavic has implemented this in the orsopy model language. You can refer to a real material and it is already contained within the database. Although this is only a very new development, and many things still to work out.

Jochen Stahn – using simple examples of the model language, but wants to put more of this info in the HDF file. Developed it initially to add some more detail in the HDF file, but it was not meant to cover everything. Artur Glavic has done further implementation.

Tom Arnold – why make the model language simpler with fewer lines. Thinks this makes it harder to be read by a human and makes it harder to see what is being done behind the scenes. We don't want it to be super complicated and long but simplifying too much can also be a problem.

Max Skoda – That is a very common argument. But ask yourself – is it beneficial to expose details (like adding the SLD of Si each time). I am suggesting to add a layer of abstraction. And this about what is the minimal amount of information needed without sacrificing details.

Tom Arnold – Worried that if we simplify too much, like abbreviations. The information or intended purpose will be lost over time

Jochen Stahn – Could save an extended version locally, so the information is not lost.

Andrew Nelson – As there are many analysis packages which do different things, we should define within ORSO what the specifications should be (thickness, SLD, roughness etc). Similar to the refnx components but have the definition within ORSO or orsopy and some demonstration code, so users can adapt it to their own specific code. ORSO would be responsible for writing the shims.

Max Skoda – agrees.

Andrew Nelson – The difficulty is how to build the constraints/bounds into the model. How do we call X datasets have common compositions etc.

Arwel - Worried that putting this information in the header of the data file could create problems, as often the sample you think you have is not the model/sample you end up with. For inexperienced users they could fall into a trap where they can never fit their data as the model is preventing them from changing the structure/model of their sample. Suggesting to disconnect the model from the datafile so there is freedom to change the sample model post experiment. Cautious about defining the model in the data file.

Max Skoda – many users struggle to fit data. This is the biggest bottleneck towards publication. Users need more than just fitted data. They need a model which is human readable and easy to interpret.

Tom Arnold - There is a counter argument to Arwel's comment – dissociating the model from the data could make it difficult keep things open and transparent.

Max Skoda – suggested you could have a "guess" model and a "final" model.

Jochen Stahn – GenX reads in the header of the data file, and outputs the fitted reflectivity and the fitted model. But it is not mandatory to define a model upfront.

Andrew Nelson – the model language is in a state where it is sufficiently simple where we have not made any bad design decisions. Rushing too quickly now could lead to problems. We should define what we expect a slab, lipid leaflet etc to look like with a well defined list of parameters, and mathematical calculation. That has not yet been done. It is better to do that now, then to start writing code. I am not sure how Artur Glavic has been progressing with that.

Paul Kienzle – If the model is included in the datafile then you can start modeling your sample as the data is being collected. This could improve experiments and throughput as you know where/when you observe the structure that you expect as data collection is happening (like mantid).

Max Skoda - Agrees. The orso format can contain multiple datasets. Can have multiple solvent contrasts in one data set. None of the catalogues allow you to add reduced data to the datafile.

Paul Kienzle - can't do it for legal / reputation reasons.

Max Skoda – but does that go against open publishing?

Jos Cooper – ESS will allow users to add reduced data to the datafile. Users will be able to write to the datafile.

Max Skoda – originally a concern was that this would have to be done manually. It is machine generated, so the user cannot upload directly

Jos Cooper – About defining what things should mean. Like a dictionary. I think it is a good idea. We should work out what things we should define in the first instance. Like, slab, lipid leaflet, magnetic layer etc.

Max Skoda – I would go a level up from that. You can define any construct, but you have to have well defined layers, and bounds. The lipid bilayer construct can be very flexible, but the input parameters can be the same. Then you can have different descriptions (like compositional description etc).

Tom Arnold – it is not quite all you need. SLDs will vary with temperature for lipids, so you also need metadata associated with your data to define what you need. So this could lead to problems.

Max Skoda – Agrees. But you have to start somewhere, and there will be limitations.

Paul Kienzle – I would like to guarantee that the profiles we are drawing is what we are calculating. Worried within refl1D the interpretation of the model and model parameters (e.g. ability to turn on/off microslabbing) is not straight forward. It is also important to ensure that the same model gives you the same output across all fitting programs.

Andrew Nelson – Once you start going from a slab to a continuous model (like a spline) then how you put everything together does matter.

Paul Kienzle – If we develop a new feature locally, how do we combine it into the standard models and fitting programs?

Max Skoda – the calculation requires slabs, and so everything in the end has to be converted to slabs. Even if you use a continuous model, the program will micro-slice it into slabs.

Arwel- To answer Paul's comment earlier. The analysis working group is making sure there is a consistency of output from all the different fitting programs.

Paul Kienzle – when a new feature is added into orso, then it should be used as a validation test for the fitting programs.

Max Skoda – we need a multipronged approach – develop tools, gather info from users. Most people here work on beamlines, so we do see what is happening and what the users need. Are there any actionable things we can start to list off?

Arwel – Prepare a list of standard sample types - lipid leaflet, lipid bilayer, slab etc

Tom Arnold – Artur Glavic has already started on this for magnetic samples, but Artur needs help with defining lipid leaflet, lipid bilayer.

Artur Glavic – joins the meeting and gives presentation of the sample builder tool he has been developing. The tool is currently available on github and is shared with the meeting participants.

Max Skoda – Thankyou Artur for the presentation. I think we are done here. All the work has been done. Presumably we can apply this model construct to anything. A clear action from this is to improve the lipid bilayer model and publicize it.

Andrew Nelson – The issue about the different definitions of samples is what we were discussing before Artur joined the meeting, so it is good to see that he is also thinking about the same things and on the same page. What happens when one class merges into another, like a free form continuous spline. That needs to be defined. Also I would not have "in solvent" but have default parameters.

Artur Glavic – Yes, there are defaults in global parameters. For the neighboring connection, I would define a new group. But I need to think about this. If you have examples, you can send them to me.

Max Skoda - I think the "in" attribute is useful.

Andrew Nelson – the difficulty comes when your head group is in one solvent, but the tail group is in other solvent.

Brian M – Is there a plan for how to define fitable parameters?

Artur Glavic – in the beginning the model language was to be used so you could see what you get before the experiment. Thinks it would be best to separate these two things – to keep the model language itself simpler.

Andrew Nelson – constraints are hard. Inequality or functional constraints are hard to define.

Max Skoda – Your GUI interprets the model and plots it very nicely (using refnx). Is it possible to generate a human readable script / fit file from this? We could include the fittable parameters in this script.

Artur Glavic - I use the resolve two-layer function which is plottable but probably not what you want to fit.

Andrew Nelson– will the aim of orsopy be to resolve to layers or classes?

Artur Glavic – you can resolve layers, then you can resolve names to the classes.

Artur Glavic – we need to decide how to parametrize the lipid leaflet, what and how many classes are needed, and if happy with the stack / sub-stack class definition.

Arwel – you only need define a lipid in terms of SLD and hydration.

Paul Kienzle – It is important to keep in mind that the SLD/thickness can vary across repeating units of a ML or lipid bilayer. Need to pull the stack apart into individual components to fit the data. How do you account for this?

Artur Glavic – I think this is more about the modelling approach. You start with a simplified model, but as you go along you have to make it more complex. I am not sure if we can include this functionality in the model language and will have to be defined by the user themselves.

Andrew Nelson – how do we innovate in this space, while allowing this to be active for users.

Artur Glavic – the things we agree on can go public, but we will keep non-approved things private. Backwards compatible.

Max Skoda – I don't think we should worry about this too much, it does not interfere with our progress, and may actually help with debugging, as the users will help us find the problems.

Jochen Stahn – in the future will use Artur's GUI to create the header information on the AMOR beamline at PSI, whenever users change samples.

Max Skoda - I will try to implement this at STFC. I think this will make a huge difference in the uptake of users of the .ort files. It is going to be a massive bonus to get the model as soon as you upload the data file.

Andrew Nelson – I would like to point out that a lot of us use python, but there are a few tools which don't use python. We are kind of excluding ourselves from those tools. With orsopy we are linking to python without the ability to change this in the future.

Final remarks:

Max Skoda – we have submitted an OSCAR (?) Grant to help us develop the model language.

Nicolo Paracini – cannot upload model data to the orso website without required permissions. Tom grants Nico the required permissions.

___

Actions

1. Meet again and finalise how we want to implement the lipid bilayer system. We need to decide how to parametrize the lipid leaflet, what and how many classes are needed, and if happy with the stack / sub-stack class definition. Arwel mentioned you only need define a lipid in terms of SLD and hydration.

2. IS to implement the sample builder tool during experiments to increase uptake, and ensure they are using the .ort format.

3. For those who cannot yet upload things to Orso, they are to email Tom Arnold their github name to be added to ORSO permissions, or email models directly to Tom and he will upload models.