

Dynamic Documentation Generation and Crowdsourcing in SasView

Brayden Miller - Albert Einstein High School
Mentor: Dr. Paul Butler



Our Facility

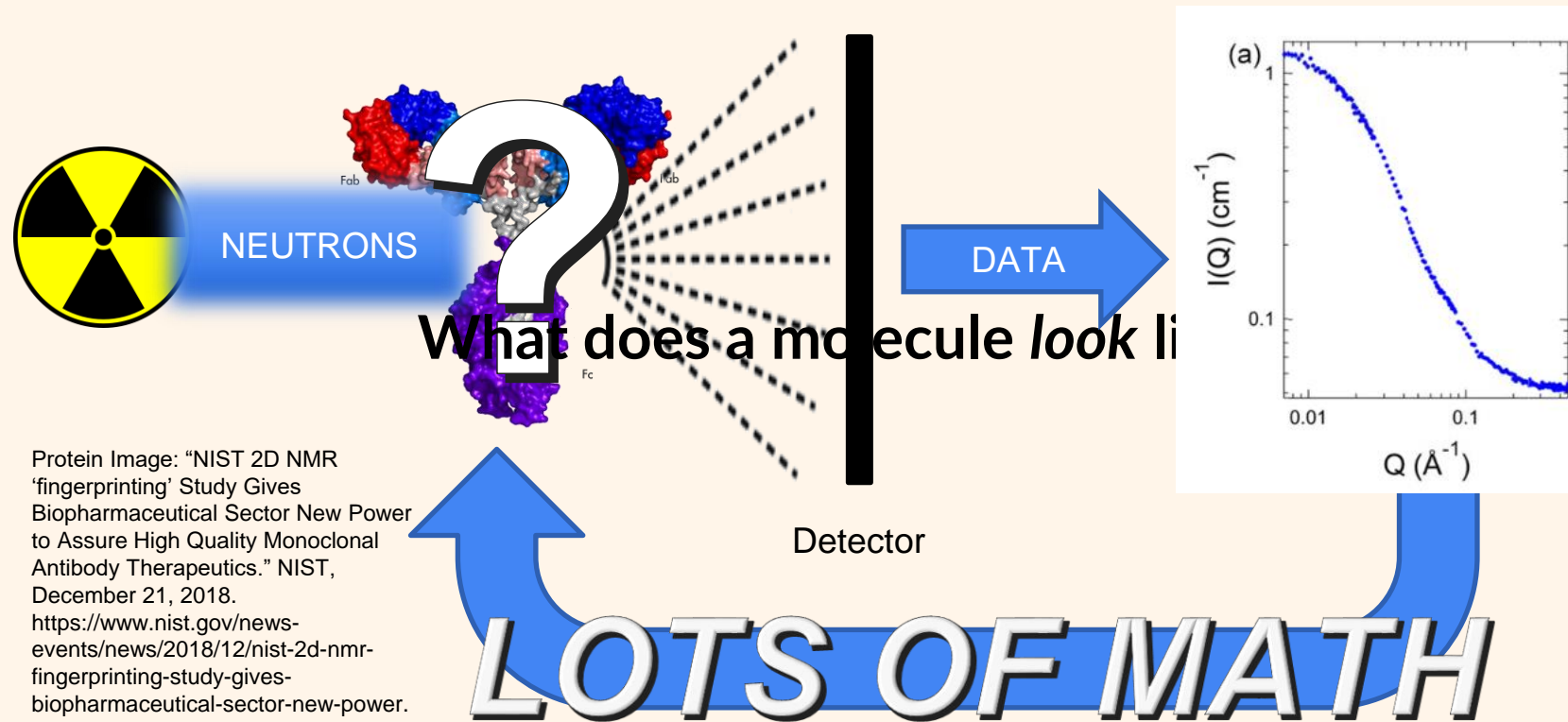
A nuclear reactor

A variety of instruments



“NCNR East Guide Hall.” *NIST*, April 17, 2017.
<https://www.nist.gov/image/20170417016guidehalljpg>.

Background: What is SANS?



Protein Image: “NIST 2D NMR ‘fingerprinting’ Study Gives Biopharmaceutical Sector New Power to Assure High Quality Monoclonal Antibody Therapeutics.” NIST, December 21, 2018. <https://www.nist.gov/news-events/news/2018/12/nist-2d-nmr-fingerprinting-study-gives-biopharmaceutical-sector-new-power>.



Yearley, Eric J., Isidro E. Zarraga, Steven J. Shire, Thomas M. Scherer, Yatin Gokarn, Norman J. Wagner, and Yun Liu. “Small-Angle Neutron Scattering Characterization of Monoclonal Antibody Conformations and Interactions at High Concentrations.” *Biophysical Journal* 105, no. 3 (August 6, 2013): 720–31. <https://doi.org/10.1016/j.bpj.2013.06.043>.

Analysis software—It does the math for us!

What makes SasView unique?

- Capable of a wide variety of analysis operations
- **Extensive documentation ~140 Pages, some ~6,000 words**
- Maintained by community



SasView

About the developer community

AI

Pro
BU



NIST

Bundesanstalt für
Materialforschung
und -prüfung

ISIS



Ahmed, Adnan. "The Bus Factor." *Tajawal* (blog), May 4, 2018.
<https://medium.com/tech-tajawal/the-bus-factor-6ea1a3ede6bd>.



SasView 5.0.6

File Edit View Tool Analysis Fitting Window Help

Data Explorer

Data Theory

Data

Load data Delete Data Select all

LOQ Standard RT2_SANS
 x_coreshell_3.txt

Send data to Fitting Batch mode Swap data

Plot Create New Append to Graph1 Help

Log Explorer

Fit panel - Active Fitting Optimizer: Levenberg-Marquardt

FitPage1

Data loaded from: LOQ Standard RT2_SANS

Model Fit Options Resolution Polydispersity Magnetism

Model

Category Model name Structure factor

Choose category... None

Options Fitting details Fitting error

f I

x^2 ...

Plot/Print Fit Help

Correlation Function
Fitting
Invariant
Pr Inversion

Kiessig Thickness Calculator
Q Resolution Estimator
Generic Scattering Calculator

Python Shell/Editor
Image Viewer
File Converter

$I = f(t_1, t_2)$

The screenshot shows the SasView 5.0.6 software interface. On the left, the 'Data Explorer' window shows two data files: 'LOQ Standard RT2_SANS' and 'x_coreshell_3.txt'. The 'Fit panel' is active, showing the 'Levenberg-Marquardt' optimizer. The 'Analysis' menu is open, with 'Fitting' selected. A red arrow points to this menu. In the bottom left, there is a diagram of a blue circular sample with two points labeled t_1 and t_2 . Below the diagram is the equation $I = f(t_1, t_2)$. A plot in the bottom right shows the intensity $I(Q)$ in cm^{-1} on the y-axis versus the scattering vector Q in \AA^{-1} on the x-axis, both on logarithmic scales. The plot shows a blue curve that starts at 10^3 and decreases to 10^{-3} as Q increases from 10^{-3} to 10^0 .

Loading Data Complete:

What does documentation look like?

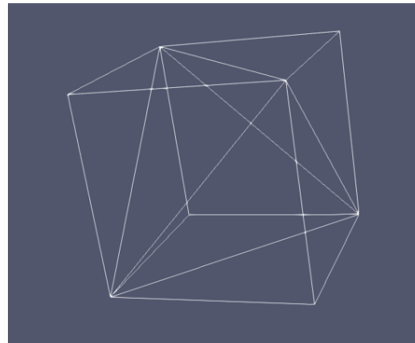
```
97 Element Type Data
98 ^^^^^^^^^^^^^^^^^^^^^
99
100 The simulation box can be described as collection of finite elements forming a mesh.
101 For example this cube is formed of five finite elements:
102
103 .. figure:: vtk_mesh_example.png
104    :align: center
105
106 Each element has an associated scattering length
107 density ( $\beta_j$ ) for the occupied space  $V_j$  and the elastic scattering
108 intensity is calculated as
109
110 .. math::
111    I(\mathbf{Q}) = \frac{1}{V} \left| \sum_j^N \beta_j \iiint_{V_j} \exp(i\mathbf{Q} \cdot \mathbf{r}_j) dV \right|^2
112
113
114 Note that the Fourier transform is calculated over each element - allowing
115 regions of space with little variation in  $\beta$  to have larger finite
116 elements, and regions of interest to have much smaller finite elements, and
117 hence more detail.
118
119 In Sasview an algorithm is implemented to calculate the Fourier transform over
120 polygons utilizing the divergence theorem as described in Maranville
121 [#MARANVILLE1]
```

What we write



Element Type Data

The simulation box can be described as collection of finite elements forming a mesh. For example this cube is formed of five finite elements:



Each element has an associated scattering length density (β_j) for the occupied space V_j and the elastic scattering intensity is calculated as

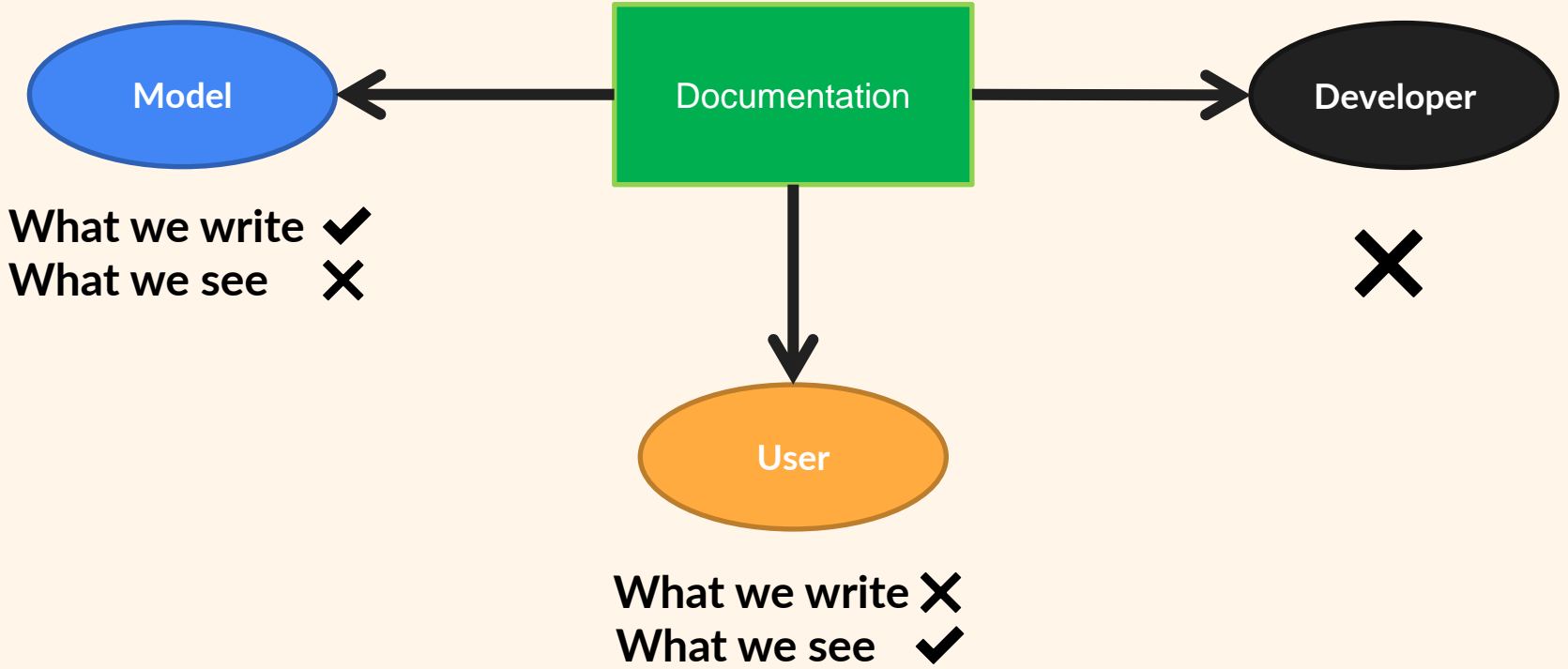
$$I(\mathbf{Q}) = \frac{1}{V} \left| \sum_j^N \beta_j \iiint_{V_j} \exp(i\mathbf{Q} \cdot \mathbf{r}_j) dV \right|^2$$

Note that the Fourier transform is calculated over each element - allowing regions of space with little variation in β to have larger finite elements, and regions of interest to have much smaller finite elements, and hence more detail.

In Sasview an algorithm is implemented to calculate the Fourier transform over polygons utilizing the divergence theorem as described in Maranville [1]

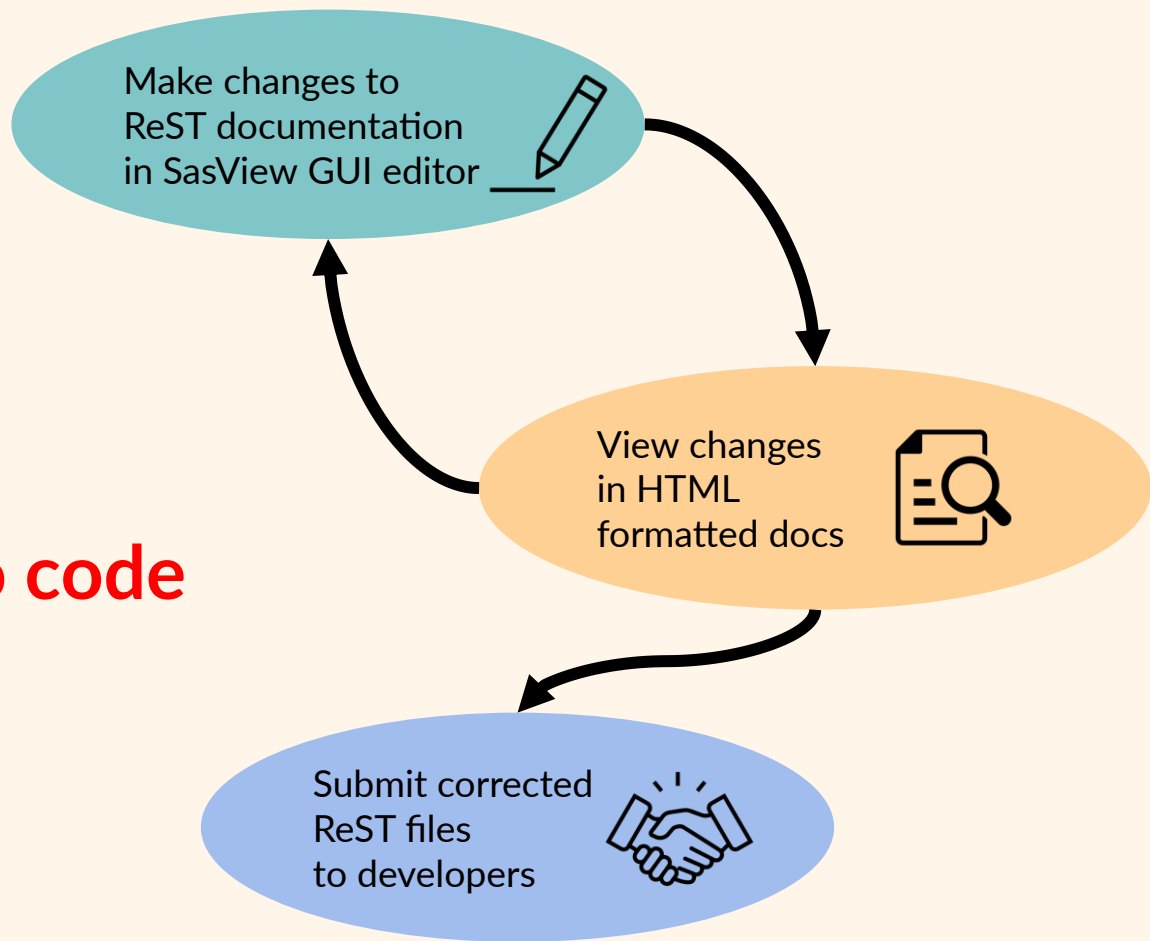
What we see

Documentation Problems:



Objective

Functional
Easy to use
Minimal changes to code
structure



Restructuring changes

SasView

Web Browser

lamellar_hg

Random lamellar phase with Head and Tail Groups

Parameter	Description	Units	Default value
scale	Scale factor or Volume fraction	None	1
background	Source background	cm ⁻¹	0.001
length_tail	Tail thickness (total = $2 \times (T + t) \times A$)	Å	15
length_head	Head thickness	Å	10
sld	Tail scattering length density	10 ¹⁰ Å ⁻³	0.4
sld_head	Head scattering length density	10 ¹⁰ Å ⁻³	3
sld_solvent	Solvent scattering length density	10 ¹⁰ Å ⁻³	6

The returned value is scaled to units of cm⁻¹ sr⁻¹, absolute scale.

This model provides the scattering intensity, $I(q)$, for a lyotropic lamellar phase where a random distribution in solution are assumed. The SLD of the head region is taken to be different from the SLD of the tail region.

Definition

The scattering intensity $I(q)$ is

$$I(q) = 2 \times \frac{\text{scale}}{2(\delta_H + \delta_T)} P(q) \frac{1}{q^2}$$

The form factor $P(q)$ is

$$P(q) = \frac{1}{q^2} [\Delta\rho_T \sin(q(\delta_H + \delta_T)) - \sin(q\delta_T)] + \Delta\rho_H \sin(q\delta_T)^2$$

where δ_T is length_tail, δ_H is length_head, $\Delta\rho_T$ is the head contrast ($\text{sld_head} - \text{sld_solvent}$), and $\Delta\rho_H$ is tail contrast ($\text{sld} - \text{sld_solvent}$).

The total thickness of the lamellar sheet is $2 \times (\text{length_tail} + \text{length_head})$. Note that in a non aqueous solvent the chemical "head" group may be the "tail region" and vice versa.

The 2D scattering intensity is calculated in the same way as 1D, where the q vector is defined as

$$q = \sqrt{q_x^2 + q_y^2}$$


SasView 5.0.5

File Edit View Tools Analysis Fitting Window Help

Data Explorer

Fit panel - Active Fitting Optimizer: Levenberg-Marquardt

No data loaded

Model Fit Options Resolution Polydispersity Magnetism

Model

Category	Model name	Structure factor
Choose category...		None

Options Fitting details Fitting error

Polydispersity Min range: 0.0005 Å⁻¹

2D view Max range: 65 Å⁻¹

Magnetism Smearing: None

χ² ...

Log Explorer

09:30:42 - INFO - SasView session started, version 5.0.5, 2022 -
09:30:42 - INFO: Python: 3.8.16 (main, May 17 2023, 17:49:16) (MSC v.1916 64 bit (AMD64))

Welcome to SasView

Old approach

Implementation

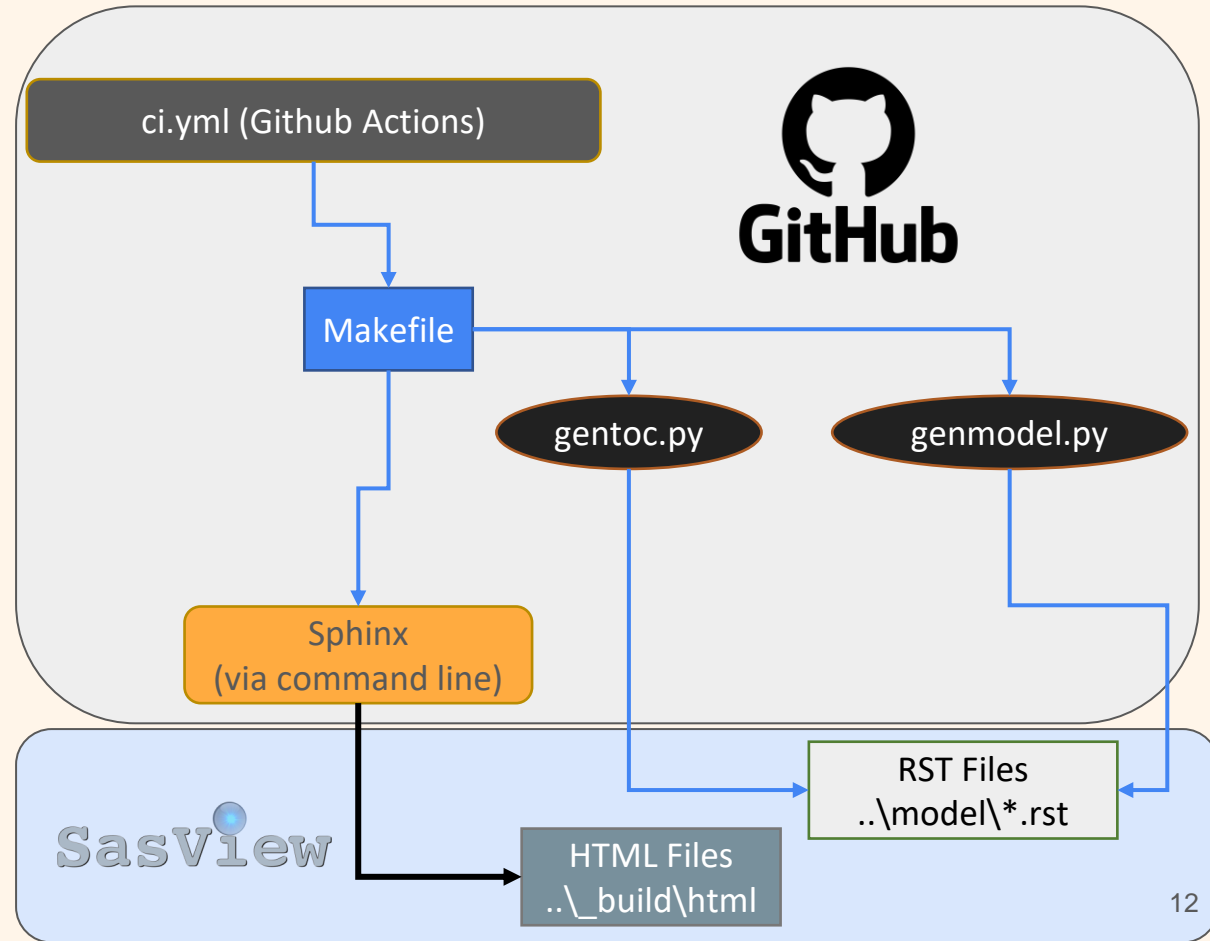
The screenshot displays a software interface with several windows. In the foreground, a 'Documentation Editor - graph_help' window is open, showing a text file named 'graph_help.rst'. The text includes instructions on how to invoke the graph menu. At the bottom of this window, three buttons are visible: 'Load file...', 'Cancel', 'Save', and 'Help'. A red arrow points to the 'Help' button.

Behind it, a 'Documentation Viewer' window shows the title '2D data averaging' and a 'Purpose' section. To the right, a plot window titled 'L.75_16.5_NIST.dat' displays a 2D color plot. A 'Slicer Parameters' dialog box is open over the plot, with a red arrow pointing to its 'Help' button.

Below the plot, another red arrow points to the 'Edit' button in the 'Documentation Editor' window. A third red arrow points to the 'Save' button in the same window.

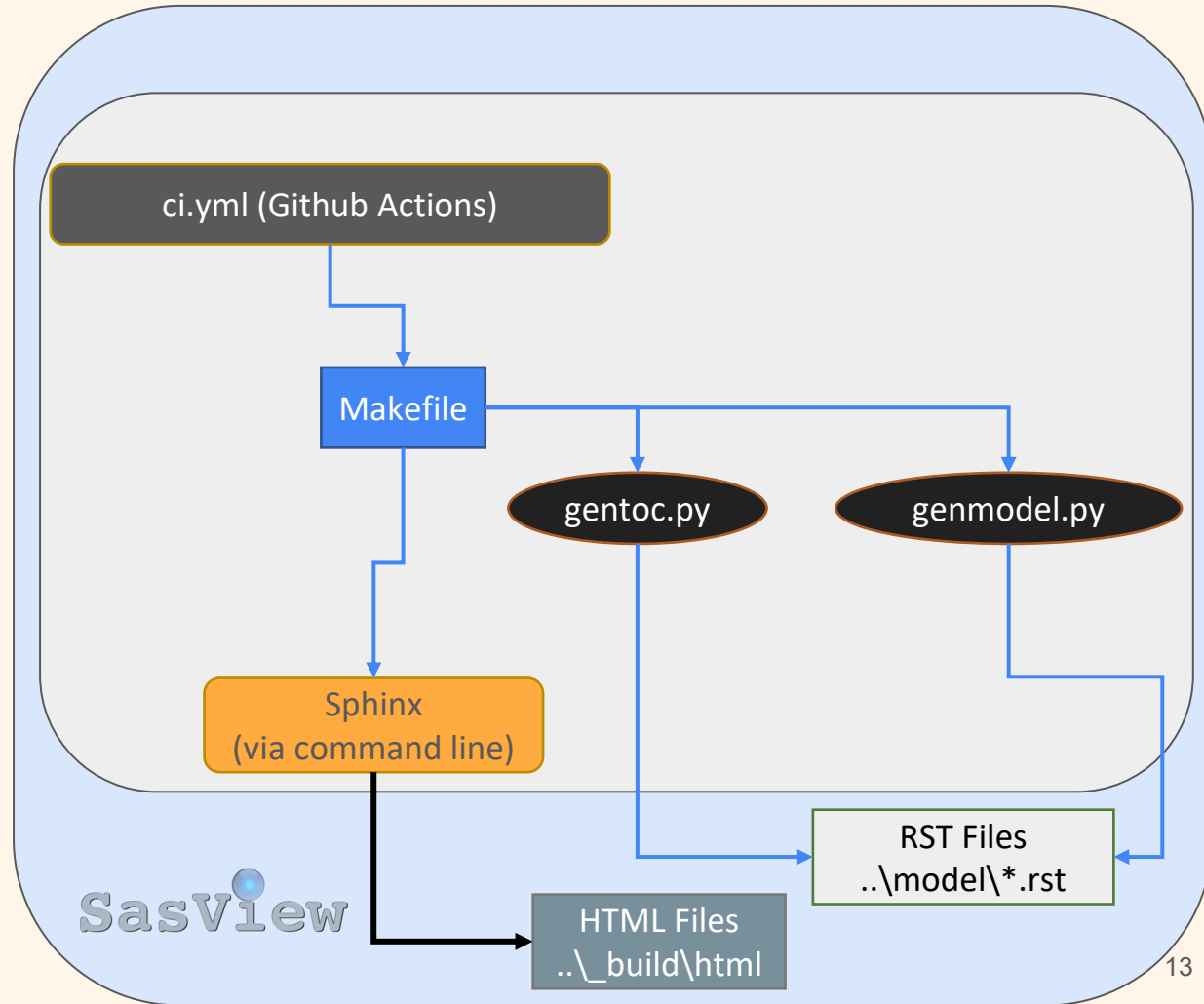
Text labels with arrows indicate the actions: 'Help Button Pressed' (pointing to the dialog's help button), 'Edit Button Pressed' (pointing to the editor's edit button), and 'Save Button pressed' (pointing to the editor's save button).

Working locally



Working locally

- Replaced functionality of current GitHub actions to compile
- Reduced runtime of GitHub processes
- Avoided using new dependencies



Summary

Accomplishments:

- Allow users to edit documentation locally
- Create opportunities for crowdsourcing SasView
- Documentation for community models is now visible

Bonus:

- Math will display regardless of browser settings

Future work:

- Automatic submission of edits to documentation
- Documentation regeneration scripts can be optimized
- Math still needs internet connection to display correctly

Special Thanks

Dr. Paul Butler

Paul Kienzle, Jeff Krzywon, Dr. Yun Liu, Dr. William Ratcliff

SHIP Program Sponsors:

Dr. Julie Borchers, Dr. Leland Harriger

And other SHIP Students

