Recent Developments in the R-Matrix Code SAMMY

D. Wiarda, G. Arbanas, J. Brown, M. Pigni, J. McDonnell, C. Chapman

Oak Ridge National Laboratory

TPR, Feb. 2023

Summary

- SAMMY releases

- Updates in the program flow

- Goals and Achievements in the Modernization Effort

SAMMY and all dependencies are freely available

SAMMY 🌐
Project ID: 6231

https://code.ornl.gov/RNSD/SAMMY

SCALE-Public
Project ID: 11984

https://code.ornl.gov/scale/code/scale-public

- The build system has been updated to automatically retrieve and build all dependencies.

- The README has been updated.

OAK RIDGE
National Laboratory

Easy SAMMY install procedure

Might have to point to QT5 (used in the testing framework):

```
export PATH=your_path_to_QT5:$PATH
```

Then from a suitable build directory:

```
cmake -D  CMAKE_INSTALL_PREFIX="desired_install_path"  ..
make –j6 install
ctest –j6
```

Configure will take some time as open scale is retrieved, but this makes the build process cleaner and ensures the correct versions are retrieved.

OAK RIDGE
National Laboratory

SAMMY Version control

- We will tag Major and Minor versions as 8.1.1 etc.

- We will tag beta releases

- The specific version will be printed in the LPT file header

- Since users now have access to pre-beta releases, we will include the hash info (pre-beta untagged releases only).

- The information gets updated automatically after downloading and configuring.

```
**********************************************************
***                                                    ***
***      SAMMY Version 8.1.pre-b1 - 893ea39b           ***
***                                                    ***
**********************************************************

Name of input file:
```

OAK RIDGE
National Laboratory

Updated SAMMY program flow

Because of the limited computer resources, SAMMY had a distributed and convoluted system to call the different modules.
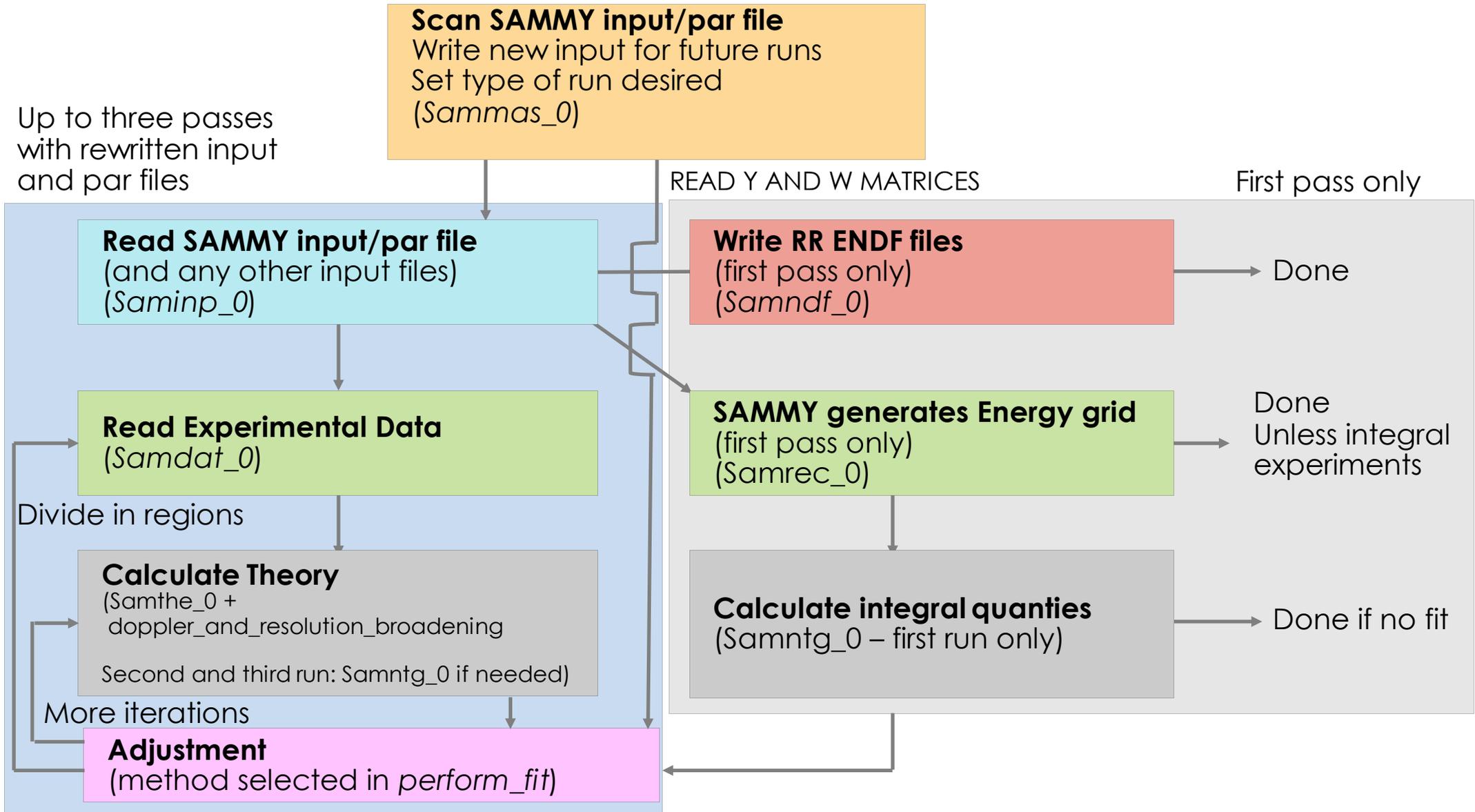
Advantage:

- Each module (by way of global parameters) determines the next step in the calculation.

Disadvantage:

- The actual flow is very hard to follow as each module directs the flow next module, i.e. lack of a governor.

- The same code appears multiple times:
  - For example: All Doppler broaden modules must point to the user desired resolution broadening function if and only if resolution broadening is wanted.

- Each module needs to have a lot of additional information regarding the user input.

OAK RIDGE
National Laboratory

SAMMY PROGRAM FLOW

Scan SAMMY input/par file
Write new input for future runs
Set type of run desired
(*Sammas_0*)

Up to three passes
with rewritten input
and par files

READ Y AND W MATRICES

First pass only

Read SAMMY input/par file
(and any other input files)
(*Saminp_0*)

Write RR ENDF files
(first pass only)
(*Samndf_0*)

Done

Read Experimental Data
(*Samdat_0*)

SAMMY generates Energy grid
(first pass only)
(Samrec_0)

Done
Unless integral
experiments

Divide in regions

Calculate Theory
(Samthe_0 +
doppler_and_resolution_broadening

Second and third run: Samntg_0 if needed)

Calculate integral quanties
(Samntg_0 – first run only)

Done if no fit

More iterations

Adjustment
(method selected in *perform_fit*)

OAK RIDGE
National Laboratory

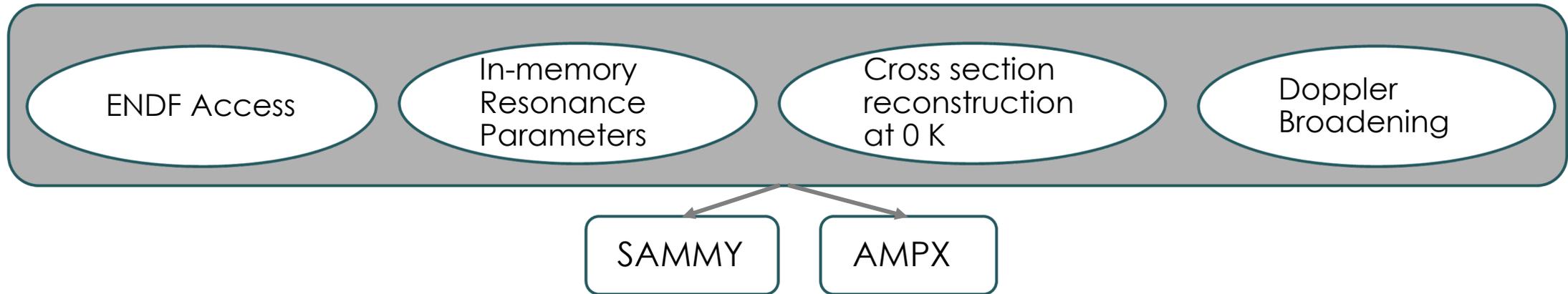# Consolidated the SAMMY program Flow

Consolidated blocks of related calls:

- One call to read input and parameter file

- One call to read data

- One call to handle theory + experimental corrections

- Flow control handled in one Fortran module

- This allows an external program to access these methods (ND10 used this already)

- But due to global parameters can't yet run in parallel

- However, data, resonance parameters and covariance information are no longer stored in global parameters.
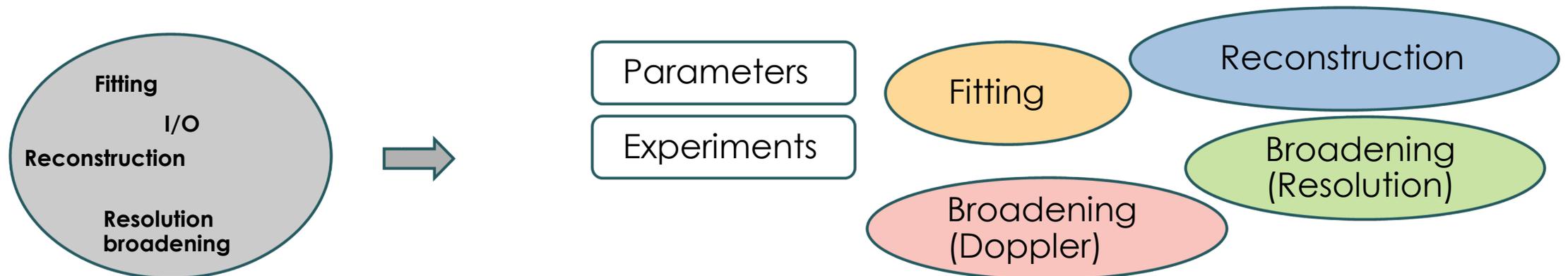
SAMMY Modernization and Maintenance goals

We plan to modernize as follows:

- Write code once and reuse

ENDF Access

In-memory Resonance Parameters

Cross section reconstruction at 0 K

Doppler Broadening

SAMMY

AMPX

- Transform SAMMY into a modular code, with independent modules with clear interfaces.

Fitting

I/O

Reconstruction

Resolution broadening

Parameters

Experiments

Fitting

Reconstruction

Broadening (Resolution)

Broadening (Doppler)

- Add new features, which is now easier as only the desired module needs to be changed.

OAK RIDGE
National Laboratory

SAMMY Modernization Accomplishments

ENDF Access

In-memory Resonance Parameters

Cross section reconstruction at 0 K

Doppler/ Resolution Broadening

Use AMPX in memory Resonance parameters + SAMMY C++ class for easier access

Made all Resolved range algorithms independent of global parameters.

Provided a class that does 0K reconstruction

Doppler broadening done, Res Broadening In progress
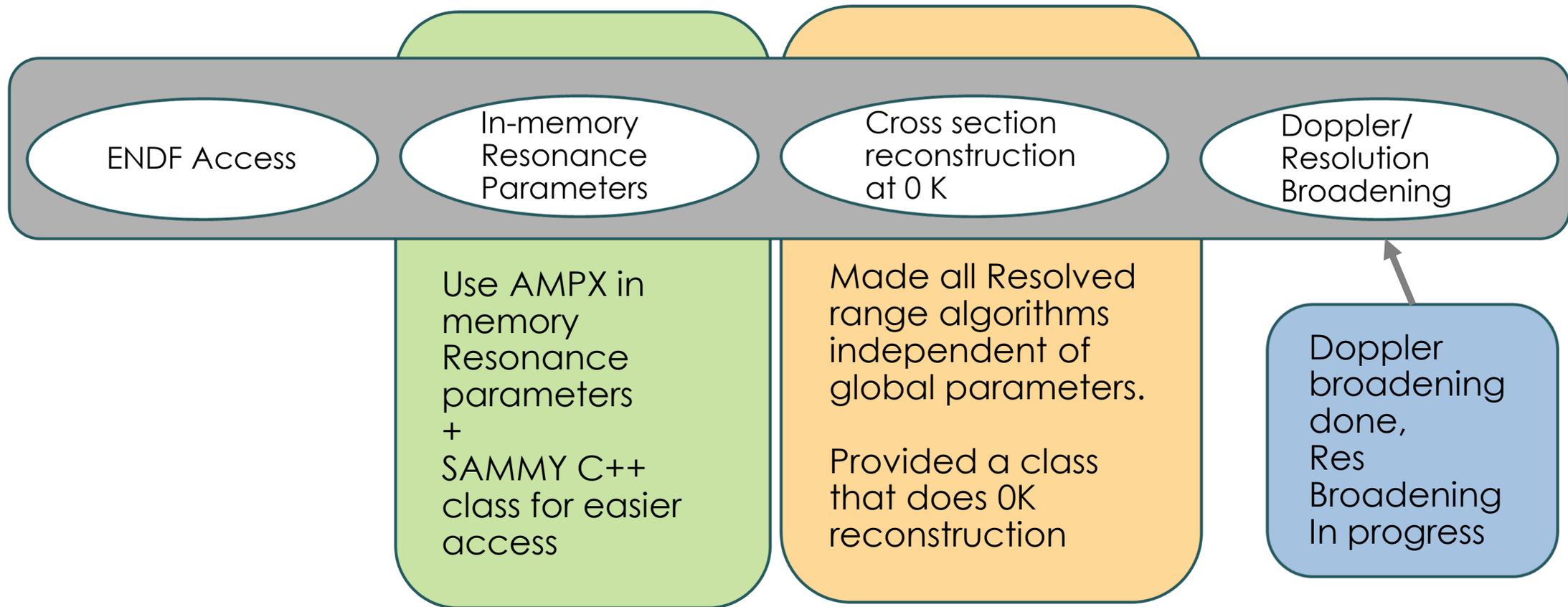
These modernizations will make adding new R-Matrix features easier as changes will be localized to the relevant part of the code as no global parameters are used. In addition, this will allow more parallelization of the code in the future.

OAK RIDGE
National Laboratory

Conversion of user documentation to LaTex

- Documentation currently is given in a Word document

- This makes updating of the documentation difficult.

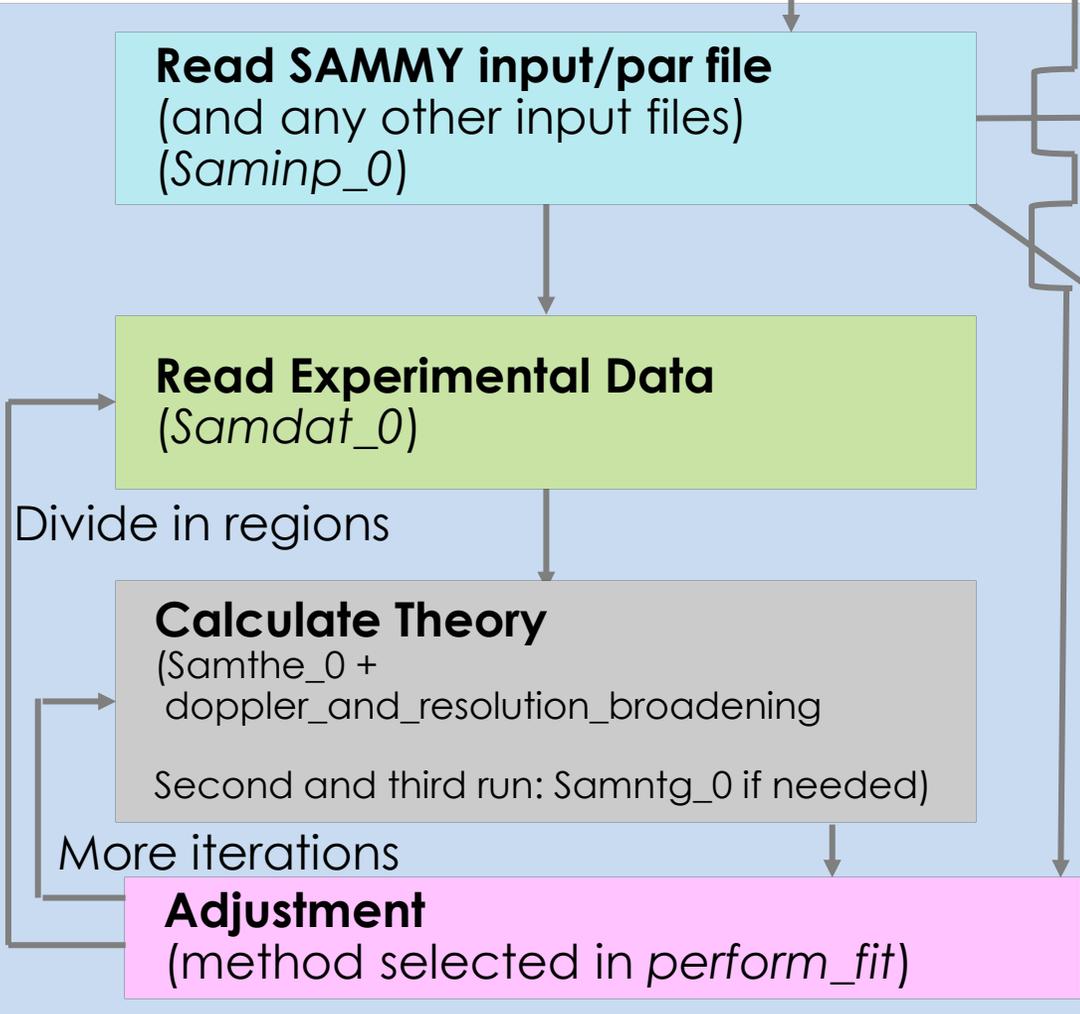- A conversion to LaTex is in progress and the ASCII files are stored in the SAMMY repository.

OAK RIDGE
National Laboratory

Conclusion

- SAMMY is available from https://code.ornl.gov/RNSD/SAMMY Versions will be tagged as needed.

- The program flow was improved to allow for easier external access to SAMMY routines.

- Modernization continued with the Doppler and Resolution broadening modules.

- Documentation updates are in progress

OAK RIDGE
National Laboratory

This work was supported by the Nuclear Criticality Safety Program, funded and managed by the National Nuclear Security Administration for the Department of Energy.

OAK RIDGE
National Laboratory

Scan SAMMY input/par file
Write new input for future runs
Set type of run desired
(*Sammas_0*)

Up to three passes
with rewritten input
and par files

READ Y AND W MATRICES

First pass only

Read SAMMY input/par file
(and any other input files)
(*Saminp_0*)

Write RR ENDF files
(first pass only)
(*Samndf_0*)

Done

Read Experimental Data
(*Samdat_0*)

SAMMY generates Energy grid
(first pass only)
(*Samrec_0*)

Done
Unless integral
experiments

Divide in regions

Calculate Theory
(Samthe_0 +
 doppler_and_resolution_broadening

Second and third run: Samntg_0 if needed)

Calculate integral quanties
(Samntg_0 – first run only)

Done if no fit

More iterations

Adjustment
(method selected in *perform_fit*)