

MOOSE IPL Extensions

(Documentation)

Cody Permann

September 2015



The INL is a U.S. Department of Energy National Laboratory
operated by Battelle Energy Alliance

DISCLAIMER

This information was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness, of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

MOOSE IPL Extensions

(Documentation)

Cody Permann

September 2015

**Idaho National Laboratory
Modeling and Simulation
Idaho Falls, Idaho 83415**

<http://www.inl.gov>

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Summary

In FY-2015, The development of MOOSE was driven by the needs of the NEAMS MOOSE-based applications, BISON, MARMOT, and RELAP-7. An emphasis was placed on the continued upkeep and improvement MOOSE in support of the product line integration goals. New unified documentation tools have been developed, several improvements to regression testing have been enforced and overall better software quality practices have been implemented. In addition the Multiapps and Transfers systems have seen significant refactoring and robustness improvements, as has the “Restart and Recover” system in support of Multiapp simulations. Finally, a completely new “Control Logic” system has been engineered to replace the prototype system currently in use in the RELAP-7 code. The development of this system continues and is expected to handle existing needs as well as support future enhancements.

CONTENTS

1. IMPROVED DOCUMENTATION.....	1
2. ROBUSTNESS IMPROVEMENTS	2

ACRONYMS

CASL – Consortium for the Advanced Simulation of LWRs

CL – Control Logic

DOE – Department of Energy

FY – Fiscal Year

MOOSE – Multiphysics Object Oriented Simulation Environment

NEAMS – Nuclear Engineering Advanced Modeling and Simulation

NQA-1 – Nuclear Quality Assurance Level 1

RELAP – Reactor Excursion and Leak Analysis Program

Error! Reference source not found.

1. Improved Documentation

To support the ever-growing base of applications built upon MOOSE, the user documentation has been expanded to cover several different media formats. Most users make use of the MOOSE wiki for code documentation and descriptions of the various systems throughout MOOSE (<http://mooseframework.org/wiki/>). This system makes use of a common “mark-down” format and is relatively easy for the developers of the framework and modules to maintain. There are however additional ways in which new developers are exposed to the workings of MOOSE. Presentation format is the next most common way to consume documentation about the framework. This format is used heavily during the MOOSE workshops held quarterly across the U.S. Finally there are several other ways to receive information about specific parts of the framework: The blog (<http://mooseframework.org/blog/>), the Doxygen based markup (<http://mooseframework.org/documentation/>), and finally the user mailing list.

The downside to all of these different formats is maintenance. The maintainers of the framework have long struggled to keep all of these separate sources up to date to reflect the evolving code-base. While the embedded Doxygen based documentation generally stays relatively up-to-date with the code changes, the wiki, blog, and training materials sometimes lag behind until application developers notice out-of-date material and request updates. To combat this issue, a new tool called “Presentation Builder” was developed to allow the developers to build documentation for all of these disparate sources in a single unified format. The Presentation Builder is able to process markdown and produce various outputs from the same sources this means that not only do we only have to maintain a single set of documentation; but that all formats of the same information is guaranteed to be identical. Figure 1 shows a slide rendered from wiki markdown. The same markdown source is used to generate online user documentation and slide-based training materials.

The tool works by processing traditional wiki markdown annotated with additional user-defined tags such as “slide breaks” or other organization tags. The user then builds the desired outline of the content they wish to see in the form of a normal “GetPot” formatted input file and runs the Presentation Builder with that input file. The markdown content is then retrieved directly from the main MOOSE framework site and is used in generating the desired format. This tool has been used to generate all training materials for the March, June and September Workshops held this year and is being used by other MOOSE-based applications as well. Plans are in place to continue augmenting the capabilities of the tool including support of different input formats such as LaTeX.

Finally, time has been spent to update documentation throughout the source code. We now include documentation checks as part of our NQA-1 peer review. Source code modifications are routinely rejected if they contain either no documentation, or documentation is not updated when code changes occur. These practices together with the new tool development have contributed greatly to the improved documentation throughout the framework.

MOOSE Code Example

Strong Form

$$\rho C_p \frac{\partial T}{\partial t} - \nabla \cdot k(T, B) \nabla T = f$$

Weak Form

$$\int_{\Omega} \rho C_p \frac{\partial T}{\partial t} \psi_i + \int_{\Omega} k \nabla T \cdot \nabla \psi_i - \int_{\partial \Omega} k \nabla T \cdot \mathbf{n} \psi_i - \int_{\Omega} f \psi_i = 0$$

Kernel
Kernel
BoundaryCondition
Kernel

Actual Code

```
return _k[_qp]*_grad_u[_qp]*_grad_test[_i][_qp];
```

15 / 423

Figure 1. Slide rendered from mark down <http://mooseframework.org/wiki/MooseTraining/Overview>

2. Robustness Improvements

FY13 and FY14 saw the design and implementation of two of the MOOSE frameworks' most powerful systems, Multiapps and Transfers. These systems enabled high-fidelity full-core reactor simulation by tying together various MOOSE-based codes. These systems are currently in widespread use throughout the INL, as well as the DOE NEAMS and CASL programs. While these systems have indeed been functional for quite some time, they were difficult to use and often failed to produce useful error messages or results when used incorrectly. In FY15, several new test cases were developed to improve the function of these new systems, catch difficult to detect failures and improve the usability of these systems.

Several enhancements to the startup, initialization, and control of the Multiapps systems have been developed and tested allowing for tighter framework control of Multiapp-enabled simulations. Better output was also engineered to aid in running and debugging simulations. Finally new dynamic loading capabilities have simplified the way users can tie new applications together without a multistep process requiring adjustments to the source code and Makefiles.

Along with the Multiapp improvements, several enhancements were made to the Transfers system. Transfers are now much better at reporting problems within the transfer rather than silently failing under various conditions. Transfers now handle non-trivial parallel domain decomposition configurations and have been made more efficient where non-scalable algorithms were in use previously. The improvements to these two systems together have made Multiapp simulations more accessible and reliable for end-use development.