

PNNL-34582

FY23 PFLOTRAN Development Support

GDSA PFLOTRAN

July 2023

Glenn E Hammond

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, express 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.** Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from
the Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062

www.osti.gov

ph: (865) 576-8401

fox: (865) 576-5728

email: reports@osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312

ph: (800) 553-NTIS (6847)

or (703) 605-6000

email: info@ntis.gov

Online ordering: <http://www.ntis.gov>

FY23 PFLOTRAN Development Support

GDSA PFLOTRAN

July 2023

Glenn E Hammond

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Acknowledgments

Funding for this research was provide by the U.S. Department of Energy Office of Nuclear Energy Spent Fuel Waste Science and Technology Program.

Acronyms and Abbreviations

CI – continuous integration

GDSA – Geologic Disposal Safety Assessment

PNNL – Pacific Northwest National Laboratory

PR – pull request

QA – quality assurance

SNL – Sandia National Laboratories

VTK – Visualization Toolkit

Contents

Acknowledgments.....	ii
Acronyms and Abbreviations.....	iii
1.0 Introduction	5
2.0 PFLOTRAN Bug Fixes	6
3.0 Updates to PFLOTRAN Software Infrastructure	9
4.0 References.....	10

1.0 Introduction

This report details the FY23 contributions to PFLOTRAN development by PNNL code developers (i.e., Dr. Glenn Hammond) under the Spent Fuel Waste Science and Technology Campaign within the US Department of Energy Office of Nuclear Energy to support the Geologic Disposal Safety Assessment (GDSA) Framework. Dr. Hammond worked closely with Rosie Leone and Peter Lichtner to refactor the implementation of the Multi-continuum Reactive Transport capability to support enhancements to the simulation process models including (1) multi-continuum implementation, (2) quality assurance (QA) and (3) bug fixes.

This report details bug fixes and updates to the software infrastructure that were implemented by Dr. Hammond in FY23. In addition to the contributions detailed in the report, Dr. Hammond met weekly with Dr. Michael Nole of Sandia National Laboratory [SNL] to strategize software design and the implementation of GDSA process models, and to discuss the acceptance of updates submitted by PFLOTRAN code developers from around the world. He contributed to and provided peer review for the SNL-led PFLOTRAN Development Milestone (M3SF-23SN010304101).

This report fulfills the GDSA PFLOTRAN Development Work Package Level 5 Milestone – *PFLOTRAN Development Support - M5SF-23PN010304062*.

2.0 PFLOTRAN Bug Fixes

Dr. Hammond fixed numerous bugs/issues reported by GDSA users, developers, and collaborators from around the world. The list below notes the major issues/bugs that directly impacted GDSA users/developers:

- Added error messaging to VTK output forcing the user to acknowledge that geometry information will not be exported, and the user must manually add that information to their file.
- Added GNU runtime error checking (-fcheck=all) to improve the comprehensiveness and robustness of continuous integration (CI) error testing.
- Removed all unused variables from the source code and added compiler directives to report errors when unused variables are present.
- Removed central difference as an option for transport (legacy code that was seldom used).
- Remove colloids from reactive transport (unused legacy code)
- Removed large automatic arrays from subroutines to eliminate GNU 11 compiler warnings. The arrays were replaced with allocatable versions that shift memory use from the stack to the heap.
- Added error messaging when the OUTPUT block is read twice.
- Added logic to only update the initial porosity with calculated porosity (based on sum of mineral volume fractions) when a flag is set, in support of dfnWorks.
- Updated the stage derived type to use PetscLogStage IDs instead of PetscInt.
- Created options_checkpoint.F90 to store all flags/options associated with checkpointing, improving the modularity of the simulation classes.
- Split factory_subsurface.F90 into three files (factory_subsurface, factory_subsurface_linkage, factory_subsurface_read) to reduce parallel compilation times, as the _read and _linkage can be compiled simultaneously.
- Removed deprecated PRESSURE, FLUX and SATURATION in favor of LIQUID_PRESSURE, LIQUID_FLUX and LIQUID_SATURATION.
- Added GAS_VISCOSITY as an output variable for GENERAL and HYDRATE modes.
- Refactored StringWriteXXX to be allocatable, eliminating trailing whitespace.
- Fixed reaction_rt_type declarations that were specified as “type” instead of “class”.
- Moved GENERAL CPR regression tests to within the HYPER_LIB conditional to prevent the test from being run when hypre is not built as a third-party library in PETSc.

- Added “make clean” operations to regression testing immediately prior to building source code to ensure that pre-existing object files are not used.
- Improved error messaging within the UnitsConversionXXX routines.
- Refactored PFLOTRAN communicator class to support more than two levels of communicators (enables infinite nesting of communicators).
- Refactored error messaging, adding print.F90 that handles all messaging funneled through PrintMsg().
- Fixed flags for turning on/off screen and file output.
- Wrapped all HDF5 open/create/close routines with error messaging that will terminate the simulation upon failure.
- Added check in DBASE routines to terminate the simulation when a database has no (zero) entries.
- Fixed memory leak in TH where a material_auxvar was not being freed.
- Added STRATA surface dataset capability where users can define material IDs between surfaces. This functionality is useful for geologic frameworks with layered media as the mapping of layers can be completed internal to PFLOTRAN.
- Updated the ExodusII conversion script, useful for converting Cubit Exodus II meshes to PFLOTRAN .h5 format.
- Refactored the writing of primal grids in HDF5 to run in parallel, useful for grids generated with dfnWorks.
- Added the ability to write verbose geochemical information for select grid cells in the domain, useful for targeted debugging of chemistry.
- Extended the default regression test time to 90 seconds.
- Added logic for force time stepping to match changes in rate and/or flux terms.
- Added verbosity to time step output when the time step is cut due to solver divergence.
- Added conditional to prevent printing of -restart file when the simulation shuts down due to failure.
- Expanded error checking on geochemistry log K's to catch undefined values for isothermal runs that are not at the reference temperature (25C).
- Added error messaging to ensure that datasets used for variables other than the datum within the hydrostatic calculation are not HDF5 format.
- Added PrintToXXX conditional around GENERAL mode output to avoid printing change in mass from each process.

- Increased verbosity of error messaging within GridLocalizeRegion() to help users better determine the source of the error.
- Added the ability to read cell IDs defining a region from a list of integers specified in the input file.
- Added a large max_step for linesearch to preserve the th_ts_1d regression test result. PETSc modified the linesearch algorithm to scale max_step and specifying a large step resolved the regression test error. Updating the regression test result was not an option as the time step size was prohibitively small.
- Fixed case of PFLOTRAN_Constants_module so that the Python script that calculates dependencies does not fail. Bug introduced by a GDSA dev.
- Added conditional to avoid check on a hydrostatic condition if the flow process model is not used for multi-continuum transport.
- Switched to the correct array of primary species names for error messaging in reactive transport.
- Added the option of using a water density of 1000 kg/m³ for the secondary continuum, allowing the use of solely molarity within the secondary continuum. This helps resolve an issue with saline density.
- Fixed a bug in regression testing where NaNs and Infs were being reported as the incorrect type of error. The error checking is now correct and more verbose reporting if the issue is NaNs or Infs.

3.0 Updates to PFLOTRAN Software Infrastructure

Dr. Hammond made the following enhancements to PFLOTRAN software infrastructure in support of GDSA applications:

- Dr. Hammond conducted weekly code reviews for pull requests (PR) submitted throughout the year. A PR is a formal request that a (Git) development branch be merged to the master branch of PFLOTRAN on Bitbucket. As outlined in the PFLOTRAN [Configuration Management Plan](#), PRs are only accepted once all continuous integration (CI) tests have successfully passed, and a lead developer has reviewed the code modifications to ensure compatibility with existing code and compliance with the PFLOTRAN Fortran coding standard.
- Dr. Hammond oversaw the upgrade to PETSc 3.18 in January and PETSc 3.19 in April (both 2023). PFLOTRAN is founded upon the PETSc library (Balay et al., 2023) that provides parallel solvers and data structures.
- Dr. Hammond updated the operating system for CI testing to Ubuntu 22. A side effect of this upgrade was an incompatibility between the MPICH 4.x installed by PETSc in GitLab Pipelines and the same version of MPICH installed on local Linux boxes. CI scripts had to be modified to install MPICH outside of PETSc configuration.
- Dr. Hammond upgraded and maintained the CI infrastructure provided by GitLab (gitlab.com/pflotran/pflotran/-/pipelines). He re-negotiated a GitLab “Open Source Program” Ultimate license which allows for PFLOTRAN developers to use GitLab repository and testing capability free of charge. CI verifies the integrity of the PFLOTRAN repository by cloning the Git repository, building third-party libraries and PFLOTRAN executables, running unit and regression tests, and comparing results against a gold standard to ensure that answers have not changed. Such testing is vital for ensuring the longevity and integrity of the code.

4.0 References

Balay, S., S. Abhyankar, M.F. Adams, S. Benson, J. Brown, P. Brune, K. Buschelman, E. Constantinescu, L. Dalcin, A. Dener, V. Eijkhout, J. Faibussowitsch, W.D. Gropp, V. Hapla, T. Isaac, P. Jolivet, D. Karpeev, D. Kaushik, M.G. Knepley, F. Kong, S. Kruger, D.A. May, L. Curfman McInnes, R.T. Mills, L. Mitchell, T. Munson, J.E. Roman, K. Rupp, P. Sanan, J. Sarich, B.F. Smith, S. Zampini, H. Zhang, H. Zhang and J. Zhang. 2023. *PETSc/TAO Users Manual*. Argonne National Laboratory, ANL-21/39 – Revision 3.19.

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354

1-888-375-PNNL (7665)

www.pnnl.gov