# FY24 Advancements in PFLOTRAN Development for GDSA Framework

## Spent Fuel and Waste Disposition

*Prepared for*
*U.S. Department of Energy*
*Office of Nuclear Energy*
*Spent Fuel & Waste Science and Technology*

*H.D. Park, D. Fukuyama, R.C. Leone,*
*M.J. Paul, A. Salazar, C.F. Madsen,*
*R.P. Rechard*

*Sandia National Laboratories*

*August 20, 2024*
**M3SF-24SN01030410**
**SAND2024-12081 R**

**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.

**DISCLAIMER**

This is a technical report that does not take into account contractual limitations or obligations under the Standard Contract for Disposal of Spent Nuclear Fuel and/or High-Level Radioactive Waste (Standard Contract) (10 CFR Part 961). For example, under the provisions of the Standard Contract, spent nuclear fuel in multi-assembly canisters is not an acceptable waste form, absent a mutually agreed to contract amendment.

To the extent discussions or recommendations in this report conflict with the provisions of the Standard Contract, the Standard Contract governs the obligations of the parties, and this report in no manner supersedes, overrides, or amends the Standard Contract.

This report reflects technical work which could support future decision making by DOE. No inferences should be drawn from this report regarding future actions by DOE, which are limited both by the terms of the Standard Contract and Congressional appropriations for the Department to fulfill its obligations under the Nuclear Waste Policy Act including licensing and construction of a spent nuclear fuel repository.

Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

# EXECUTIVE SUMMARY

The Spent Fuel & Waste Science and Technology (SFWST) Campaign of the Office of Spent Fuel & Waste Disposition of U.S. Department of Energy Office of Nuclear Energy (DOE-NE) is conducting research and development on geologic disposal of spent nuclear fuel (SNF) and high-level nuclear waste (HLW). This report describes fiscal year 2024 accomplishments in the Geologic Disposal Safety Assessment (GDSA) PFLOTRAN Development work package, which is charged with developing subsurface simulation software for postclosure performance assessment of deep geologic disposal of SNF and HLW.

PFLOTRAN is the multi-physics simulation engine that is being developed for use in of the GDSA Framework, a software toolkit for probabilistic performance assessment of deep geologic disposal systems. As such, PFLOTRAN has been developed to include important components of the radionuclide source term model such as waste form inventory, radionuclide decay/ingrowth, waste form degradation, and radionuclide mobilization. The radionuclide source-term model is coupled to the flow and transport models. These flow and transport models include processes such as multiphase non-isothermal flow with advective, diffusive, and dispersive radionuclide transport through either a single- or dual-continuum formulation of porous media while considering chemical reactions and radionuclide decay and ingrowth.

In fiscal year 2024 (FY24), new capabilities were added to PFLOTRAN, which will be included in PFLOTRAN Version 6.0 to be released in October 2024. Key capabilities developed include implementation of a buffer erosion/canister corrosion model, updates to the Nuclear Waste Transport mode, completion of the quasi-implicit wellbore model, enhancements to fully implicit salt transport option, and refinements to the dual continuum model.

The buffer erosion/copper corrosion model (BECC) simulates (1) buffer erosion from the flow of water through a fracture intersecting a deposition hole or emplacement tunnel, and (2) the resulting localized corrosion of a waste package over time. Two versions of the BECC are planned: a virtual model that does not change buffer porosity/permeability and a coupled model that does change buffer porosity/permeability. The latter model requires re-running fluid flow conditions at each timestep. In FY24, the virtual BECC model was implemented first and an example is presented.

Nuclear Waste Transport (NWT) mode is designed to simulate the transport of radioactive nuclides and their reaction under all saturation conditions including dry-out conditions in porous media. Dry-out can be caused by heat from radioactive decay or gas generation from material degradation processes driving liquid away from waste packages. In FY24, new regression tests were created that coupled NWT mode with the four fluid flow modes in PFLOTRAN[1] to monitor NWT mode integrity as future developments are undertaken. Also, some functionality from the Used Fuel Disposition (UFD) Decay (UFD Decay) process model was included in NWT mode in FY24. Work in FY25 will continue to add functionality from UFD Decay to NWT mode.

In FY24, a quasi-implicit wellbore process model was implemented into PFLOTRAN to simulate drilling intrusion scenarios using GDSA Framework. This model addresses the complex interactions that occur when a wellbore is drilled into a repository, potentially breaches containment barriers, and alters flow and transport in the subsurface environment. The one-dimensional wellbore grid is calculated separately from the three-dimensional repository system model to eliminate the need for mesh refinement around the wellbore, which significantly decreases computation time. In addition, a well model can be easily added to an existing model.

In FY22, implicit transport of a solute was added to PFLOTRAN's GENERAL mode, which includes a solute/salt mass conservation equation in addition to the conservation equations for water, air, and energy.

In FY24, the capability to define both soluble and insoluble solid phase materials was added to the Solute Salt option for GENERAL mode. This capability is useful for simulating insoluble geologic and engineered materials adjacent to soluble host rock (e.g., halite) in a repository. An example of halite dissolution and subsequent density driven flow is presented. In addition, an example is presented that shows how the capability can be used to simulate heat flux and fluid flow in a system undergoing creep closure.

The dual continuum process model in PFLOTRAN underwent several refinements in FY24. The ability to have a variable number of secondary continuum cells within the dual continuum block was added. This ability is useful when simulating a model with a heterogenous effective diffusion coefficient or to turn off the dual continuum model in some regions of the model domain. The dual continuum model and UFD Decay were applied to a generic spent nuclear fuel repository in fractured crystalline rock as described in the international research project DECOVALEX.

Support activities included the development of a calculation archive, debugging the nuclear criticality function in PFLOTRAN's waste form process model, PFLOTRAN documentation, support for existing GDSA conceptual models[2], and PFLOTRAN maintenance. Outreach included a PFLOTRAN introductory short course.

Performance enhancements included proposing a roadmap for using graphic processor units (GPUs) with PFLOTRAN. GPU usage in PFLOTRAN would take advantage of new high-performance computing architectures that incorporate GPUs to accelerate a portion of the computing. Two questions are addressed here. First, does the underlying problem lend itself to GPU computing? And, if so, how can the existing software be adapted for this use? We concluded that to use GPUs, we first should develop a vectorized PFLOTRAN based on standard Fortran and then, if warranted, develop a PFLOTRAN vector version that uses available GPU software in non-standard language extensions or libraries for Fortran.

Repository systems analysis in generic host rocks continues to be a key driver for developing new capabilities in PFLOTRAN. In FY24, a major focus was transitioning from small-scale or prototype computational models to full-scale, safety-assessment models for uncertainty quantification and sensitivity analysis (UQ/SA), aiming to complete a production-ready version of PFLOTRAN. The UQ/SA work, along with the criticality scenario work in a dual-purpose canister, has significantly tested these new capabilities this year.

Specifically, efforts to simulate a large-scale model, involving thousands of individual waste packages running on thousands of processors (i.e. solving for millions of degrees of freedom), introduced new challenges. While PFLOTRAN has successfully handled system-scale simulations in the past, the introduction of new capabilities and use cases lead to discovery of code bugs in the software that were undetected in the prototype and small-scale tests. Initial attempts at some of these advanced simulations encountered issues that were not replicable in smaller test-scale models, presenting a unique debugging challenge. Much of the debugging had to be done manually in the high-performance computing environment for these large-scale simulations. Additionally, some of the debugging process took months of effort and required extensive communication between modelers and developers. The Agile Software Development framework adopted in FY23 greatly facilitated this communication, enabling more efficient problem-solving and iteration.

[1] PFLOTRAN simulation modes: GENERAL—two-phase, non-isothermal, miscible fluid flow, RICHARDS—single-phase fluid flow, WIPP—two-phase, isothermal, immiscible fluid flow, and TH—heat transport coupled to single-phase fluid flow.

[2] GDSA conceptual models include multi continuum model, waste form process model, buffer swelling model, material transform capability, and other geologic disposal concept numerical or mathematical models.

# ACKNOWLEDGEMENTS

This report includes contributions from the following co-authors:

David Fukuyama: Nuclear Waste Transport mode, Implicit Salt Transport mode, Waste Form process model, and Wellbore model support

Rosie Leone: Buffer Erosion/Copper Corrosion model and Dual Continuum model

Calvin Madsen: Calculation Archive

Heeho Park: Quasi-Implicit Wellbore model, Software Infrastructure, and Outreach

Matthew Paul: Graphic Processing Unit Roadmap

Alex Salazar: Nuclear Waste Transport mode, Comparison Regression Test of Used Fuel Disposition Decay mode to Nuclear Waste Transport mode

Rob Rechard: Compilation and organization of the FY 24 milestone report.

The authors thank Kris Kuhlman and Lisa Bigler of Sandia National Laboratories (SNL) for their thoughtful technical review and Prasad Nair from U.S. Department of Energy Office of Nuclear Energy (DOE NE), for discussions, oversight, and guidance on topics addressed in this report.

# CONTENTS

Page

# FIGURES

**Figure** **Page**

# TABLES

**Tables** **Page**

# NOMENCLATURE

| | |
|---|---|
| BECC | buffer erosion/copper corrosion (model) |
| CTN | calculation tracking number |
| DCDM | dual continuum disconnected matrix |
| DOE-NE | U.S. Department of Energy Office of Nuclear Energy |
| ECPM | equivalent continuous porous medium |
| EDZ | excavation damaged zone |
| GDSA | Geologic Disposal Safety Assessment |
| GENERAL | two-phase, non-isothermal, miscible fluid flow fully implicit mode of PFLOTRAN |
| GIRT | global implicit reactive transport mode of PFLOTRAN |
| GPU | graphics processing unit |
| HLW | high-level radioactive waste |
| MPI | message passing interface (communication standard used in parallel processing) |
| NWT | nuclear waste transport mode of PFLOTRAN |
| PETSc | Portable Extensible Toolkit for Scientific Computation |
| RICHARDS | single-phase, saturated/unsaturated fluid flow mode of PFLOTRAN |
| SFWST | Spent Fuel & Waste Science and Technology |
| SKB | Svensk Karnbranslehantering AB (Swedish Nuclear Fuel and Waste Management Company) |
| SNF | spent nuclear fuel |
| TH | thermo-hydro flow mode (heat transport coupled to single-phase fluid flow) |
| UFD decay | used fuel disposition decay mode of PFLOTRAN |
| UQ/SA | uncertainty quantification/sensitivity analysis |
| WIPP | two-phase, isothermal, immiscible fluid flow mode for use at Waste Isolation Pilot Plant repository in southeastern New Mexico |

# 1. INTRODUCTION

## 1.1 Campaign Background

The Spent Fuel & Waste Science and Technology (SFWST) Campaign of the Office of Spent Fuel & Waste Disposition of U.S. Department of Energy Office of Nuclear Energy (DOE-NE) is conducting research and development on geologic disposal of spent nuclear fuel (SNF) and high-level nuclear waste (HLW). A high priority for SFWST Campaign disposal research and development is disposal system modeling (DOE 2012, Table 6; Sevougian et al., 2019; Sassani et al., 2021). The Geologic Disposal Safety Assessment (GDSA) PFLOTRAN Development work package of the SFWST Campaign is charged with developing subsurface simulation software for postclosure performance assessment of deep geologic disposal of SNF and HLW. This report fulfills the requirements of the GDSA PFLOTRAN Development work package (SF- 24SN01030410) Level 3 Milestone – Recent Advancements in PFLOTRAN Development for the GDSA Framework, M3SF-24SN01030410.

## 1.2 PFLOTRAN Description

PFLOTRAN (Hammond and Frederick 2016; Lichtner et al., 2013; Lichtner and Hammond, 2012; Hammond et al., 2011) is an open source, massively parallel multiphase flow and reactive transport simulator designed to leverage high-performance computing infrastructure for subsurface earth systems applications. PFLOTRAN solves coupled systems of nonlinear partial differential equations describing non-isothermal multiphase flow and reactive transport in porous media. Parallelization is achieved through domain decomposition using the Portable Extensible Toolkit for Scientific Computation (PETSc) (Balay et al., 2013, 2022). PETSc provides a flexible interface to data structures and solvers that facilitates the use of parallel computing. PFLOTRAN is written in modern Fortran and leverages state of the art Fortran programming (Fortran classes, pointers to procedures, etc.) to support its object-oriented design. The modular design of the code allows developers to integrate a custom set of process models and time integrators for simulating different surface and subsurface multi-physics processes. PFLOTRAN employs a single, unified framework for simulating multi-physics processes on both structured and unstructured grid meshes (i.e., no code needs to be duplicated to calculate multi-physics processes on structured or unstructured meshes). PFLOTRAN requires a small set of standard third-party libraries e.g., PETSc (above), MPI (message passing interface), BLAS/LAPACK (basic linear algebra subprograms/linear algebra package), HDF5 (hierarchical data format version 5), METIS/PARMETIS (serial and parallel software packages for partitioning unstructured meshes). Both the unified structured/unstructured framework and the small number of third-party libraries greatly facilitate usability.

PFLOTRAN is a multi-physics engine integrated into the GDSA Framework, a software toolkit for probabilistic postclosure safety assessment of deep geologic disposal systems for spent nuclear fuel and high-level radioactive waste (Figure 1-1). As such, PFLOTRAN has been developed to include important components of the radionuclide source term model, which includes waste form inventory, radionuclide decay/ingrowth, waste form degradation, and radionuclide mobilization. The radionuclide source-term model is coupled to the flow and transport models. These flow and transport models include processes such as multiphase non-isothermal flow with advective, diffusive, and dispersive radionuclide transport through either a single- or multi-continuum formulation of porous media while considering chemical reactions and radionuclide decay/ingrowth. Radionuclide output from the transport model can provide input to a biosphere model to estimate dose to a human receptor.

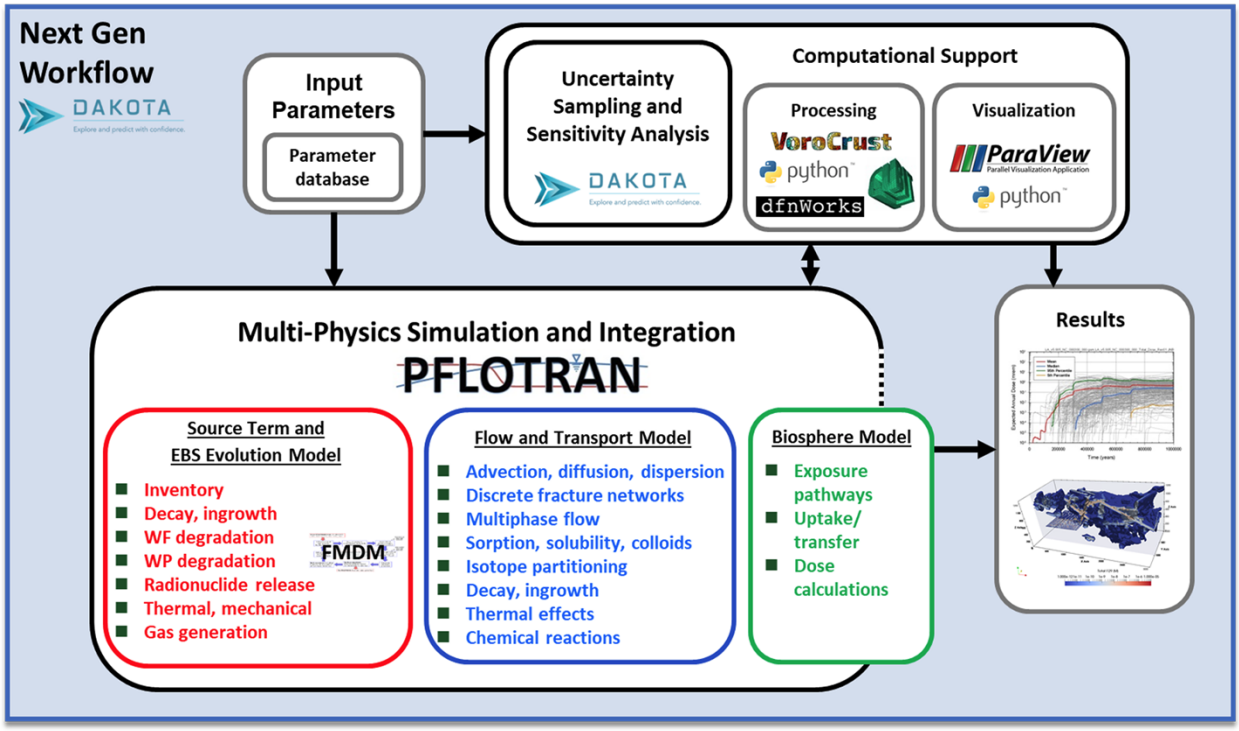



**Figure 1-1. GDSA Framework Schematic (Mariner et al., 2023).**

PFLOTRAN integrates modeling components of a generic geologic disposal system (Figure 1-2) through either fully coupled simulation (e.g., heat and multiphase fluid flow) or sequential coupling (e.g., fluid flow followed by transport of radionuclides). Its outputs may also be passed to a downstream code for additional analysis (e.g., biosphere processes).

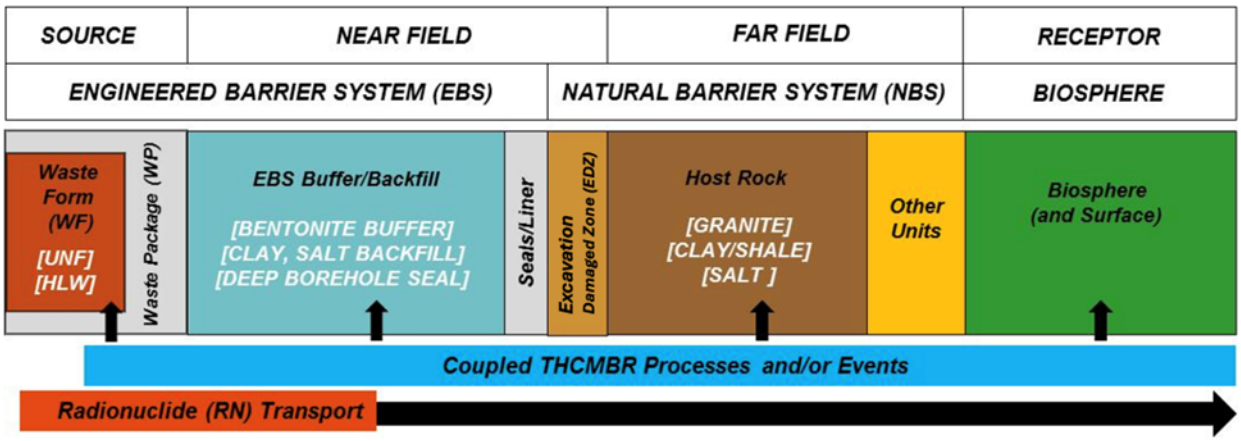



**Figure 1-2. General subsystems and components of a generic geologic disposal system.**

Four of PFLOTRAN's flow modes have been developed and used for simulating deep geologic disposal systems. The three flow modes are used in simulation of generic deep geologic disposal systems: RICHARDS mode (single-phase, saturated/unsaturated fluid flow), TH (Thermo-Hydro) mode (single-phase fluid flow coupled to heat transport), and GENERAL mode (two-phase, miscible, non-isothermal fluid flow). The fourth, WIPP flow mode (two-phase, immiscible, isothermal fluid flow), was developed for use in performance assessment calculations for the Waste Isolation Pilot Plant (WIPP).

## 1.3 PFLOTRAN Applications Beyond GDSA Framework

PFLOTRAN has been employed on petascale leadership-class computing resources (e.g., Jaguar at Oak Ridge National Laboratory). Users can select combinations of process models and flow/transport coupling to tailor the analysis of the porous media under study. PFLOTRAN has been used to simulate thermal-hydrological-chemical processes at the Nevada National Security Site (Mills et al., 2007), multiphase $CO_2$-$H_2O$ injection for carbon sequestration (Lu and Lichtner, 2007), $CO_2$ leakage within shallow aquifers (Navarre-Sitchler et al., 2013), uranium fate and transport at the Hanford 300 Area (Chen et al., 2013), surface-subsurface flow coupling (Wu et al., 2021), geothermal systems (Alt-Epping et al., 2021), and multiphase liquid-gas-gas hydrate systems (White et al., 2020).

## 1.4 PFLOTRAN Development and Maintenance in FY24

In fiscal year 2024 (FY24), new capabilities were added to PFLOTRAN, which will be included in PFLOTRAN Version 6.0 to be released in October 2024. Key capabilities developed include refinements to the buffer erosion/canister corrosion process model (Section 3.1), updates to Nuclear Waste Transport (NWT) mode (Sections 3.3 and 3.2), completion of the quasi-implicit wellbore process model (Section 3.5) and enhancements to fully implicit salt transport process model option (Section 3.6). Support activities included the development of a calculation archive (Section 2.2), debugging the nuclear criticality function in the WASTEFORM process model (Section 4.3.5), PFLOTRAN documentation, support for existing GDSA conceptual models, and PFLOTRAN maintenance (Appendix A). Performance enhancements included proposing a roadmap for using graphic processor units (GPU) with PFLOTRAN (Section 5). Outreach included a PFLOTRAN introductory short course (Section 6).

Repository systems analysis in generic host rocks continues to be a key driver for developing new capabilities in PFLOTRAN. In FY24, a major focus was transitioning from small-scale or prototype computational models to large-scale, safety-assessment models for uncertainty quantification and sensitivity analysis (UQ/SA). The UQ/SA work, along with the criticality scenario work in a dual-purpose canister, has significantly tested these new capabilities this year.

Efforts to simulate a field-scale model (i.e., millions of degrees of freedom), involving thousands of individual waste packages running on thousands of processors, introduced new challenges. While PFLOTRAN has successfully handled system-scale simulations in the past, the introduction of new capabilities and use cases led to the discovery of code bugs that were undetected in the prototype and small-scale tests. Initial attempts at some of these advanced simulations encountered issues that were not replicable in smaller test-scale models, presenting a unique debugging challenge. This process required extensive manual debugging in the high-performance computing environment.

Some of the debugging process took months of effort and required extensive communication between modelers and developers. The Agile Software Development framework greatly facilitated this communication. Our developers successfully completed all the necessary bug fixes for the UQ/SA, dual-purpose canister nuclear criticality analysis, and repository system analysis work in time for the modelers to run their simulations and achieve their goals for the year. Sections 3.4.3 (Debugging NWT), 3.5.1 (Debugging Well Model), and 4.3 (General Debugging) provide detailed explanations of the debugging process, including complications, issues, and corrections.

# 2. SOFTWARE INFRASTRUCTURE

This year saw significant progress towards a production-ready version of PFLOTRAN. The software was tested with a wide range of parameters derived from Latin Hypercube Sampling, which introduced new and challenging conditions. Running this diverse set of parameters led to some unexpected outcomes, such as high-temperature dry-out of waste packages in a high-pressure, low-permeability, fully saturated shale host rock. These cases required additional investigation to determine whether the simulations were pushing the physical limits or if there were any code bugs that needed to be addressed.

The debugging of PFLOTRAN this year demonstrated the efficiency of the Agile software development method, based on Jira. Enhanced communication between modelers and developers facilitated the resolution of issues and allowed for thorough discussions within the team. The application of pair programming, a key component of the Agile method, was particularly effective in tackling and solving very difficult problems.

Additionally, we developed a Python-based calculation archive in FY24 to preserve simulation inputs, code versions, authors, and associated published papers. The calculation archive will ensure the replicability of these simulations, which will safeguard the integrity and continuity of our work.

## 2.1 Jira Report

PFLOTRAN development activities are tracked in Jira through "issues," which encompass the development of new capabilities, bug fixes, record creation, and code maintenance. In the Agile software development method, each issue is referred to as a "task." Developers volunteer to take responsibility for these tasks during a "sprint," a two-week period in which the tasks must be completed.

Each sprint includes short meetings to plan the sprint (distributing tasks), check progress mid-sprint, and review the sprint's progress to close out completed tasks and discuss any tasks that were not completed on time. From August 2023 to July 2024 (past 365 days), 141 issues were created, and 129 issues were resolved (Figure 2-1). Appendix A categorizes these issues into four general categories (1) PFLOTRAN maintenance, (2) support for existing GDSA conceptual models, (3) refinements to the buffer erosion/canister corrosion model, and (4) enhancements to salt mode.

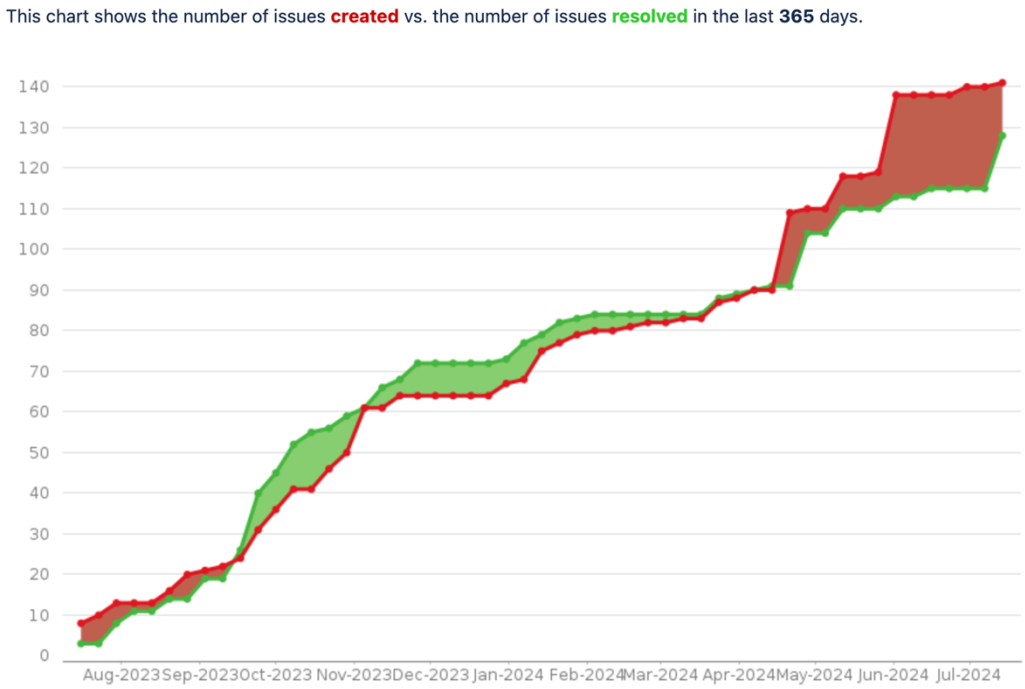



**Figure 2-1. FY24 Issues Created versus Issues Resolved.**

## 2.2 Calculation Archive

The archival of simulation experiments is important to establish reproducibility in calculations over several versions of PFLOTRAN. Having a central hub to store prior and current calculations used in reports and experiments allows for team members to easily access calculations, as well as determine exactly how calculations were run, and what software and version were used. Therefore, it is important to develop an easy-to-use method of archiving these calculations, as well as establish a central server on-site to store these archived calculations.

The calculation archive must have the following properties, ease of use, unique identifiers for each stored calculation, a secure storage location, unit testing and error handling, and support for long-term development. Each property is considered during development of the calculation archive, resulting in software that provides the user an easy-to-use tool to archive calculations.

### 2.2.1 Specifications and Functions

The calculation archive was created to run on a Sandia National Laboratories (SNL) on-site server. This on-site server runs the archival code, as well as stores the calculations. The code is written in Python with an object-oriented design. When archiving calculations, the user provides both required and optional metadata. This information is handled as an input file. The input file is parsed by the archival code and is the only required input for the calculation archive. The input file allows for users to easily add information about the calculation to be stored in the archive. The required information includes the path to the user's calculation to be archived, the author names of the calculation, the email of the point of contact, the title of the calculation, and the software and version used for that calculation. This information is mandatory and is stored in the archive with the calculation. The information which is archived onto the server includes the files in the folder specified by the user in the input file with a few exceptions. To decrease the size of

archived calculations, files which are detected as simulation output files are ignored and not archived on the server.

Each calculation is assigned a Calculation Tracking Number (CTN). The CTN is used as the directory name for a calculation in the archival server. This CTN has the format YYMMDD-$$$$$$$-##, where the three sections correspond to the date in year-month-day format, alphabetical characters, and a number starting at 01 and increasing for every repeated CTN. The user may specify the date or the alphabetical characters in the input file, otherwise the current day and a random character string are used. This CTN ensures a unique identifier is assigned to each calculation, which may be easily referenced to later find the archived calculation.

The calculation archive appends each archived calculation into an easily readable database text file. This database contains the calculation CTN, title, and point of contact. This database allows users to easily find CTNs corresponding to a title or point of contact.

The calculation archive also allows the user who created the archived calculation (or the overall archive owner) to remove a calculation if a mistake is discovered, by entering the CTN. However, the creator of an archived calculation may not remove archived calculations created by others.

Test cases were created to ensure the correctness of the software with further developmental changes. These test cases were created with the open-source testing module `doctest` in Python. They ensure the robustness of error handling, proper feedback to the user, and that information is archived.

### 2.2.2 Using the Calculation Archive

The script which runs the archive is located on the same server which stores the calculations. A user's calculations are often initially stored on their local machine or other locations which are not the same as the on-site server. Therefore, the user accesses the calculation archive software through bash scripts provided to them.

This script, as well as the archive code, provides useful error handling and output for the user to determine what the code is doing, and shows any input errors. The bash script will use the user's input file to copy their calculation over to the remote server using the `rsync` command. The Python script is then run on the remote server using Secure Shell Protocol (`ssh`). This script provides the user the main two functionalities, to archive a calculation, and to remove an archived calculation.

### 2.2.3 Documentation and Repository

The documentation for this code was generated using Sphinx, a powerful open-source documentation generator that converts `reStructuredText` files into various output formats, such as HTML and PDF. Sphinx allows the user to easily update documentation as the code is further developed. The documentation provides details on the purpose of the calculation archive, how to use the calculation archive, and how it works. The documentation and code for the calculation archive is found in a GitLab repository at Sandia National Laboratories (SNL).

# 3. PROCESS MODEL DEVELOPMENT

## 3.1 Buffer Erosion/Copper Corrosion Process Model

The buffer erosion/copper corrosion (BECC) process model simulates (1) buffer erosion from the flow of water through a fracture intersecting a deposition hole or emplacement tunnel, and (2) the resulting localized corrosion of a waste package over time. The BECC process model placed in PFLOTRAN is based on the models developed and used by Sweden (SKB—Neretnieks et al., 2017) and Finland (Posiva, 2013) for the Forsmark and Olkiluoto repositories, respectively. The development of this capability is ongoing; Section 3.3 describes progress in FY24 toward implementing the BECC process model as well as future plans.

### 3.1.1 Buffer Erosion

The rate of buffer loss during a given time step depends on the ionic strength of the groundwater, groundwater velocities in the intersecting fracture, fracture properties, and smectite behavior. The model of Neretnieks et al. (2017), adopted here, involves three primary processes: 1) buffer intrusion, 2) buffer erosion by seeping water, and 3) buffer erosion by sedimentation. The equations used by PFLOTRAN for each erosion mechanism are described in Nole et al. (2023). When groundwater ionic strength exceeds a user-defined threshold, e.g., 4 mM, the buffer loss rate from the deposition hole is only calculated from buffer intrusion. When ionic strength is below the ionic strength threshold, the overall buffer loss rate is the sum of all buffer loss processes modeled: buffer intrusion and buffer erosion at the rim by water flow and sedimentation.

The SKB buffer erosion model is a zero-dimensional model, whereas the repository model domain may be up to three-dimensional. The repository model domain may have cells containing buffer that lose buffer as predicted by a zero-dimensional buffer erosion model. Two distinct buffer erosion models will be implemented: a virtual buffer erosion model and an additive buffer cell erosion model. The SKB model is referred to as a virtual buffer erosion model because it runs in the waste package domain but does not remove any material from the flow model domain. The additive buffer cell erosion model will run at each buffer cell location where there is a qualifying intersecting fracture. It will be consistent with the virtual buffer erosion model but will also remove buffer from the flow model domain, and, thereby, enhance the flow of water to a waste package borehole and the flow of corrosive reactants to the waste package surface. Therefore, PFLOTRAN will need to iterate the fluid flow simulation.

### 3.1.2 Implementation of Virtual Buffer Erosion Model

In FY24, the implementation of the virtual BECC process model was completed. The virtual BECC was implemented into the waste form process model in PFLOTRAN within the CANISTER_DEGRADATION_MODEL block. The amount of virtual buffer eroded is calculated at each time step. Once the critical volume of buffer erodes, the copper corrosion model starts. When the canister corrodes to the depth of the canister wall thickness, the waste package breaches, and canister vitality is set to zero. The model calculates the time it takes for the buffer to erode and copper to corrode. At each time step the model prints out the amount of (virtual) buffer eroded, copper corroded, and times of the buffer and copper failure. Below is an example input sequence in the waste form process block that calls for the buffer erosion/copper corrosion model in a domain with one fracture and one waste package:

```
WASTE_FORM_GENERAL
 WASTE_FORM
  …
  FRACTURE_ANGLE 1.5708 # radians
```

```
  FRACTURE_APERTURE 0.0001 # m
  WATER_VELOCITY  41 # m/yr
  ION_CONCENTRATION 1.946  # mol/L
/

MECHANISM CUSTOM
  …
  CANISTER_DEGRADATION_MODEL
    CANISTER_MATERIAL_CONSTANT 1500.
    BUFFER_EROSION_COPPER_CORROSION
      …
    /
  /
END #MECHANISM CUSTOM

END_WASTE_FORM_GENERAL
```

### 3.1.3 Example Problem of Virtual BECC Model of a Waste Package

An example of the virtual BECC model applied to one waste package with constant flow through one fracture is implemented in a 9 × 9 m$^2$ two-dimensional domain (Figure 3-1). The fracture aperture, angle, groundwater velocity and buffer and canister properties are input into the virtual BECC model (Appendix B). In this example, when the copper canister thickness is corroded completely, the waste package breaches, and a tracer is released into the model domain at the location of the waste package.

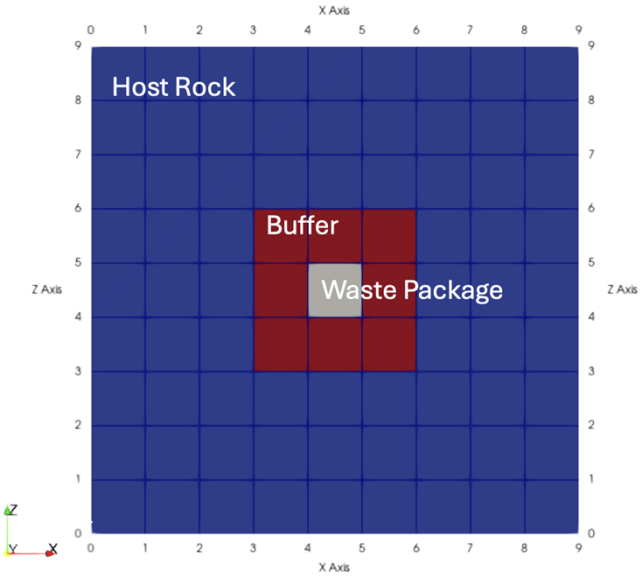



**Figure 3-1. Two-dimensional 9 × 9 m$^2$ material domain for BECC example.**

The amount of buffer eroded over time is plotted in Figure 3-2. The buffer erosion reaches the canister at 4073.3 years and then copper corrosion begins. The copper corroded over time is plotted in Figure 3-3. The copper canister breaches at 1,187,783 years, at which time the tracer is instantly released into the model domain at the waste package grid cell. Figure 3-4 shows the concentration of the tracer in the model domain 917 years later at 1,188,700 years. The virtual buffer erosion model does not remove any buffer cells in the domain, but the planned buffer cell erosion model will remove buffer cells with intersecting fractures.

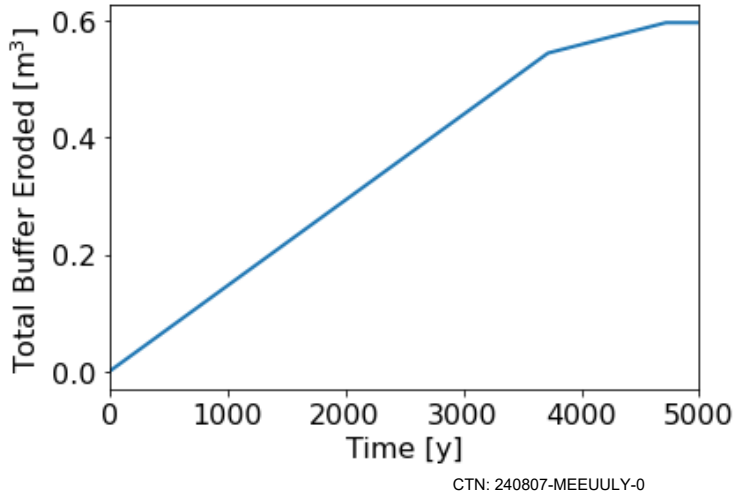



**Figure 3-2. Total buffer eroded over time until erosion reaches canister at 4073.3 years.**

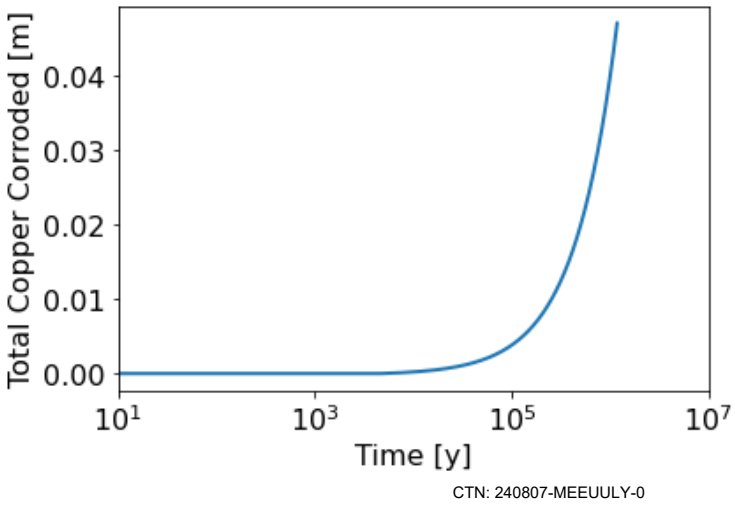



**Figure 3-3. Total canister corrosion over simulation time until canister with 0.047-m wall thickness breaches at 1,187,783 years.**

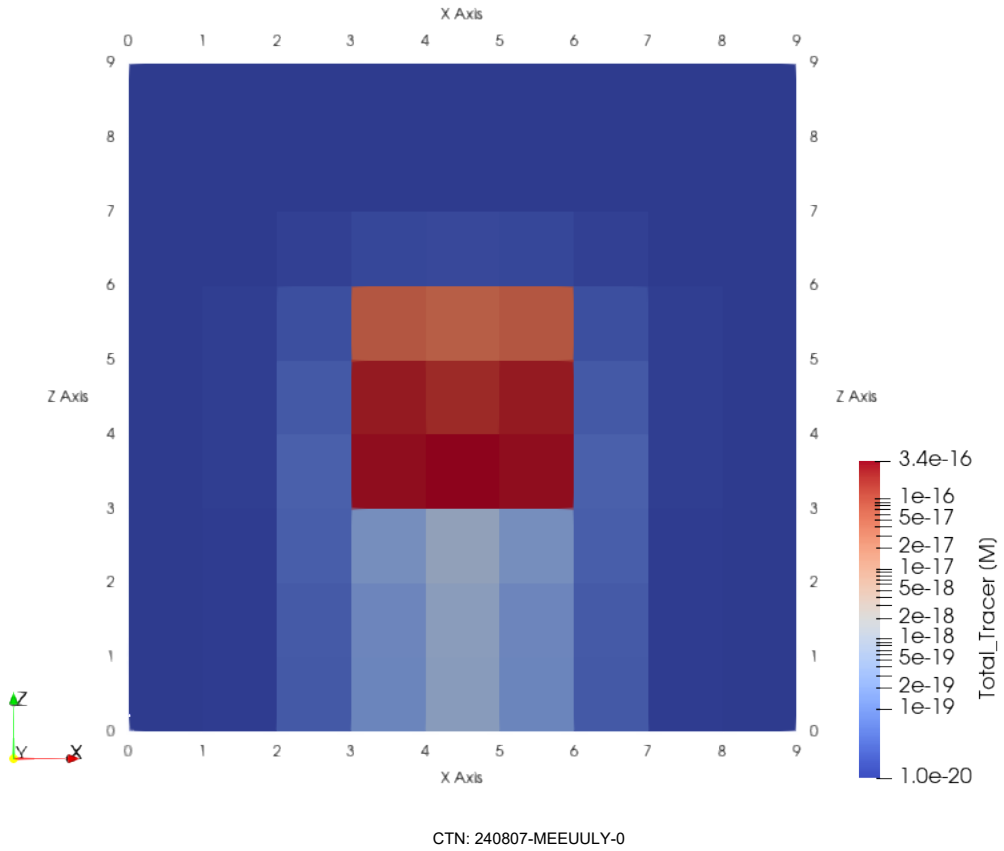



**Figure 3-4. Concentration of non-sorbing, non-decaying tracer in model domain 917 years after canister breach at 1,188,700 years.**

### 3.1.4 Plans for Future Work

Future work for the virtual BECC model includes applying it to a generic crystalline repository reference case simulation with multiple waste packages in a complex fracture network. The plan is to incorporate BECC model into the general corrosion capability of PFLTORAN. These steps will also entail creating quality assurance tests.

## 3.2 Migration of UFD Decay Process Model Functions to NWT Mode

This section summarizes the developments to the NWT mode to include functionality from the Used Fuel Disposition (UFD) Decay process model. The UFD Decay process model (Mariner et al., 2016) simulates decay and ingrowth of radionuclides and their partitioning (on an elemental basis) among the aqueous, sorbed, and solid phases. It is used in conjunction with PFLOTRAN's Global Implicit Reactive Transport (GIRT) mode, whose mass balance equations are formulated in terms of aqueous concentration (mol/L). NWT mode, which was originally developed for PFLOTRAN simulations of the Waste Isolation Pilot Plant, has mass balance equations formulated in terms of bulk concentration ($mol/m^3_{bulk}$), a feature that enables calculations to continue through periods of complete dry out.

The motivation to include the UFD decay process model into NWT mode is to reduce the time step lagging error previously observed in the UFD decay and GIRT mode sequential coupling (Nole et al., 2022). In its current implementation, GIRT mode cannot resolve these time step lagging issues. Given that NWT mode is the capability that GDSA will adopt in the future, it is necessary to add this capability to NWT

mode. By integrating UFD decay into NWT mode, we can achieve a fully coupled capability, effectively eliminating time step lagging numerical errors in the simulation.

### 3.2.1 Enabling Complete Dry-Out with NWT Mode

NWT mode (Nole et al., 2023) is a reactive transport mode based on immiscible flow, where species are transported via advection and diffusion assuming equilibrium chemistry of precipitation/dissolution and sorption/desorption. Transport is assumed to occur in the liquid phase. NWT mode is designed to allow transport simulations to continue under dry conditions or saturation below residual in porous media caused by heat generation driving liquid away from near waste packages or gas generation by corrosion and degradation of waste package materials.

In reference case simulations, heat-generating waste can cause water to convert to steam and the pore space to dry. In GIRT mode, issues arise because the primary dependent variable of concentration (mol/kg-water) requires a solvent (water) phase to be present. As water vaporizes, the saturation of the pore space decreases. To retain the same dissolved quantity, the concentration in GIRT mode must increase. As the water saturation approaches zero, the component concentration increases towards infinity, causing errors in the numerical solution.

To avoid simulation errors caused by solving for aqueous concentration, NWT mode uses bulk concentration as a primary dependent variable (mol/m$^3$-bulk) (Frederick et al., 2020). The total concentration is first solved, then the moles within each grid cell are partitioned between the dissolved, precipitated, and sorbed phases, and the radioactive decay/ingrowth is implicitly solved within each phase (see Eq. 3-1 below). Using total concentration as the primary dependent variable removes the dependency on water saturation for a solution. The general solution methodology is described in Frederick (2021).

### 3.2.2 Regression Test Coverage

For PFLOTRAN version 5.0, 14 regression tests use the UFD Decay process model, of which 13 tests employ GIRT mode and one employs NWT mode. While some of these tests employ no flow, others use the GENERAL, TH, RICHARDS, and WIPP flow modes.

As of now, UFD Decay will remain in PFLOTRAN to support GIRT-related analyses that are still in progress. However, in terms of porting capability, the migration of UFD Decay functionality to NWT mode should cover the thirteen GIRT tests (out of 230) currently in use. Adding these analog tests will also provide additional coverage to NWT mode for the various flow modes. Furthermore, eleven of the UFD Decay tests use the waste form process model, which would offer additional coverage for that process model when the analog tests are added. Currently, NWT mode is only used in 22 regression tests.

### 3.2.3 User Input Migration

Merging functionality from UFD Decay process model to the NWT mode involved modifying the NUCLEAR_WASTE_CHEMISTRY card to accommodate additional keyword options. In NWT, common keywords are shown in the example input in Table 3-1.

**Table 3-1. Chemistry Input deck for NWT mode.**

| Chemistry Input Deck for NWT Mode |
|---|
| NUCLEAR_WASTE_CHEMISTRY<br><br>SPECIES<br>NAME U234L<br>SOLUBILITY 1.000d+20 # [mol/m^3-liq] |

**Chemistry Input Deck for NWT Mode**

```
  PRECIP_MOLAR_DENSITY 3.861d+04    # [mol/m^3-mnrl]
  ELEMENTAL_KD       0.000d+00    # [m^3-water/m^3-bulk]
 /

 SPECIES
  NAME             TH230L
  SOLUBILITY         1.000d+20    # [mol/m^3-liq]
  PRECIP_MOLAR_DENSITY 3.861d+04    # [mol/m^3-mnrl]
  ELEMENTAL_KD       1.000d-05    # [m^3-water/m^3-bulk]
 /

 RADIOACTIVE_DECAY
 # [1/sec]
  8.983245d-14 U234L -> TH230L
  0.d0        TH230L
 /
 OUTPUT
  ALL_SPECIES
  ALL_CONCENTRATIONS
  MINERAL_VOLUME_FRACTION
 /

 TRUNCATE_CONCENTRATION 1.0d-20

END

CONSTRAINT constraint_inventory_initial
 CONCENTRATIONS
  U234L    5.940705d-04   VF    # [m^3-mnrl/m^3-void]
  TH230L   2.717432d-06   VF    # [m^3-mnrl/m^3-void]
 /
END
```

In UFD Decay, common keywords are shown in the example input in Table 3-3.

**Table 3-2. Chemistry input deck using the UFD Decay process model.**

**Chemistry Input Deck for UFD Decay**

```
CHEMISTRY

 PRIMARY_SPECIES
  Np-237
  U-233
  Th-229
 /

 MINERALS
  Np-237(s)
  U-233(s)
```

**Chemistry Input Deck for UFD Decay**

```
  Th-229(s)
 /

 MINERAL_KINETICS   # Placeholder (not currently in use)
  Np-237(s)
   RATE_CONSTANT 0.d0
  /
  U-233(s)
   RATE_CONSTANT 0.d0
  /
  Th-229(s)
   RATE_CONSTANT 0.d0
 /

 OUTPUT
  TOTAL
  Np-237
  U-233
  Th-229
 /

 DATABASE ./ufd-decay.dat
END


CONSTRAINT groundwater
 CONCENTRATIONS
  Np-237    1.e-20       F
  U-233     1.e-20       F
  Th-229    1.e-20       F
 /
 MINERALS
  Np-237(s) 0.d0 1.d0
  U-233(s) 0.d0 1.d0
  Th-229(s) 0.d0 1.d0
 /
END

UFD_DECAY
 ELEMENT Np
  SOLUBILITY 4.07d-9
  KD
   soil1 3.5d5
  /
 /
 ELEMENT U
  SOLUBILITY 3.16d-14
  KD
   soil1 6.1d5
  /
```

**Chemistry Input Deck for UFD Decay**

```
 /
 ELEMENT Th
  SOLUBILITY 7.94d-11
  KD
   # soil1 2.5d6
   DATASET kd_Th_soil1.txt soil1 # use time-dependent Kd dataset
  /
 /
 ISOTOPE Np-237
  ELEMENT Np
  DECAY_RATE 1.03d-14
  DAUGHTER U-233 1.d0
 /
 ISOTOPE U-233
  ELEMENT U
  DECAY_RATE 1.38d-13
  DAUGHTER Th-229 1.d0
 /
 ISOTOPE Th-229
  ELEMENT Th
  DECAY_RATE 2.78d-12
 /
END
```

Additionally, for UFD Decay, an interface with the material transform process model can be used to modify $K_d$, a measure of how strongly a contaminant adheres to the solid phase compared to remaining in the liquid phase, during a simulation such as in the smectite-to-illite model (Nole et al., 2023). The smectite-to-illite model is a forward model that tracks the smectite fraction in a region (those containing clay minerals) based on the known kinetics of the mineral phase transition to illite, which is determined by the potassium concentration, activation energy, and temperature. Table 3-3 shows a material transform object defined for the material "soil1," where the $K_d$ three elements from Table 3-3 can be modified with linear functions with the specified constants. During each time step, material transform evaluates the temperature for each cell. If the temperature is above a threshold temperature, the fraction of illite is modified based on the previously recorded fraction for that cell. In turn, the illite content is used in the linear models to shift the $K_d$ values for each element.

**Table 3-3. Material transform input used with the UFD Decay process model to modify $K_d$.**

**Material Transform Input Deck with UFD Decay**

```
#========================= material transformations ===========================
MATERIAL_TRANSFORM_GENERAL
 MATERIAL_TRANSFORM mt_soil1
  ILLITIZATION
   ILLITIZATION_FUNCTION GENERAL
    THRESHOLD_TEMPERATURE 2.50000d+1 C
    EA                7.00000d+3 cal/mol
    FREQ              8.44907d-6
    K_CONC            2.16000d-3 M
    K_EXP             2.50000d-1
    SMECTITE_INITIAL     9.75000d-1
```

**Material Transform Input Deck with UFD Decay**

```
      SMECTITE_EXP         5.00000d+0
      SHIFT_PERM LINEAR    5.00000d+2
      SHIFT_KD
        Np LINEAR  0.10000d+0 # Np from UFD Decay
        U  LINEAR  0.10000d+0 # U from UFD Decay
        Th LINEAR  0.10000d+0 # Th from UFD Decay
      /
    END
  END
 END
END
```

### 3.2.4 Analytical Differences

While changes have been discussed regarding the PFLOTRAN input deck, there are modifications that are needed to implement the UFD Decay capability into the NWT mode due to the fundamental governing equation difference. In NWT mode and UFD Decay, the total bulk mass ($M$) of species $i$ is given by the sum of aqueous (A), precipitated (P), and sorbed (S) phase partitions $\alpha$ which are assumed to be in equilibrium. The unit of $M_i^\alpha$ is mol/m$^3$-bulk (Frederick et al., 2020):

$$M_i^\alpha = M_i^A + M_i^P + M_i^S \quad \text{Eq. 3-1}$$

for aqueous (A), precipitate (P), and sorbed (S) phases. In turn,

$$M_i^A = \phi S_l C_i^A \quad \text{Eq. 3-2}$$

$$M_i^P = \phi S_p C_i^P \quad \text{Eq. 3-3}$$

$$M_i^S = \phi S_l K_d C_i^A \quad \text{Eq. 3-4}$$

Where $C_i^\alpha$ is the α phase concentration of species $i$ (mol/m$^3$-liquid), $\phi$ is the media porosity, $S_l$ is the liquid saturation, $S_p$ is solid pore volume fraction, and $K_{d,e}$ (Eq. 3-5) is the linear sorption coefficient of element $e$ (m$^3$-water/m$^3$-bulk) (and where the units of $K_{d,e}$ differ from those used in GIRT because of the primary variable units used in NWT mode (as mentioned again in Section 3.4.3).

The UFD Decay process model performs radionuclide isotope decay, ingrowth, and phase partitioning for the simulation of a nuclear waste repository. The following steps describe the general methodology of UFD Decay:

- User specifies the aqueous concentration of a species $C_i^A$.
- The sorbed concentration of species $i$ is determined with the aqueous concentration and the sorption distribution coefficient $K_{d,e}$ of the element ( $e$ ):

$$C_i^S = C_i^A \cdot K_{d,e} \quad \text{Eq. 3-5}$$

- The masses of all phases are summed in each time step according to Equation Eq. 3-1.
- Decay/ingrowth is evaluated for all isotopes $i$ during the time step using the Bateman equations (either explicitly for up to three generations or using an implicit solution).

- After the decay/ingrowth of all isotopes, the total mass of each isotope is then partitioned back into aqueous, sorbed, and precipitated phases. Mole fractions ($X$) are calculated to determine the fraction that each isotope contributes to total element mass:

$$X_i^S = \frac{M_{i,total}}{M_{e,total}} \quad \text{Eq. 3-6}$$

- The elemental aqueous concentration is calculated using the total mass of the element, $K_d$, the liquid saturation ($S_l$), porosity ($\phi$), liquid density ($\rho_{H2O}$), and grid cell volume ($V$).

$$C_e^A = \frac{M_{e,total}}{\left(1 + \frac{K_{d,e}}{\rho_{H_2O}\phi S_l}\right) \cdot V\phi S_l} \quad \text{Eq. 3-7}$$

- If the aqueous concentration of the element exceeds the elemental solubility limit ($C_e^*$), then the aqueous elemental concentration is made equal to the elemental solubility limit, and the remaining elemental mass is then partitioned between sorbed and precipitated phases.

$$C_e^S = C_e^A \cdot K_{d,e} \quad \text{Eq. 3-8}$$

- If the aqueous concentration was set to the solubility limit, then the mass of precipitate is calculated using Eq. 3-1 (i.e., by moving $M^P$ to the left hand side), and then the precipitate concentration is calculated using the precipitated molar volume of the element ($V^{mnrl}$) as follows:

$$C_e^P = \frac{M_e^P V_e^{mnrl}}{V} \quad \text{Eq. 3-9}$$

- Isotope concentrations are calculated from the partitioned elemental concentrations by multiplying by the isotope mole fractions.

$$C_i^\alpha = X_i C_e^\alpha \quad \text{Eq. 3-10}$$

- How are sources and sinks treated in UFD Decay?
    - In UFD Decay, sources and sinks are included in a residual equation incorporating the parent/daughter concentration solution from the Bateman equations (i.e., the implicit solution).
    - The Bateman solution for nuclide $m$ is written as follows, where $\lambda$ are decay constants and $\gamma$ is a branching ratio of the parent nuclide ($m$ - $1$), which were shown as "1.0" in the Table 3-2 example:

$$\frac{dC_m(t)}{dt} = -\lambda_m C_m(t) + \gamma\lambda_{m-1} C_{m-1}(t) \quad \text{Eq. 3-11}$$

    - For time step $k$ of width $\Delta t$ and iteration $p$, if the solution of the Equation Eq. 3-11 can be symbolized as $c_k$, then the residual equation $f$ can be defined as follows, where $R$ is the source/sink term and the left portion is the accumulation term:

$$f(c^{k+1,p}) = \frac{c^{k+1,p} - c^k}{\Delta t} - R(c^{k+1,p}) \quad \text{Eq. 3-12}$$

    - The Jacobian ($J$) matrix incorporating this residual is defined as:

$$J_{ij} = \frac{\partial f_i(c^{k+1,p})}{\partial c_j^{k+1,p}} \quad \text{Eq. 3-13}$$

- This matrix is used in the following system of equations, where δ is a Newton step:

$$J(c^p)\delta = -f(c^{k+1,p}) \quad \text{Eq. 3-14}$$

- When Equation Eq. 3-14 is solved by Newton's method, the concentration of the time step can be updated:

$$c^{k+1,p+1} = c^{k+1,p} + \delta c^p \quad \text{Eq. 3-15}$$

To mitigate simulation errors caused by solving for aqueous concentration, NWT mode solves for the total bulk mass of species in phases. The general methodology in NWT mode is described as follows (Frederick, 2021):

- The total bulk mass of species is governed by Equation Eq. 3-1 above, where the different phases are assumed to be in equilibrium. The time-dependent governing equation in NWT for the mass of species *j* in phase α is described in Eq. 3-19 in terms of the phase velocity (***u***), dispersion tensor ***D***, and the source/sink terms for influx ($Q$) and decay ($R$).

$$\frac{\partial}{\partial t}\sum_{\alpha} M_j^{\alpha} + \nabla \cdot \left(\boldsymbol{u}^{\alpha} - \boldsymbol{D}_j^{\alpha} \cdot \nabla\right) M_j^{\alpha} = \sum_{\alpha} Q_j^{\alpha} + \sum_{\alpha} R_j^{\alpha} \quad \text{Eq. 3-16}$$

- However, in NWT mode, the aqueous phase (A) is modeled as the only mobile phase, while the sorbed and precipitated phases are stationary with no diffusion or dispersion. Therefore, Eq. 3-19 is modified as follows:

$$\frac{\partial}{\partial t}\sum_{\alpha} M_j^{\alpha} + \nabla \cdot \left(\boldsymbol{u}^{A} - \boldsymbol{D}_j^{A} \cdot \nabla\right) M_j^{A} = \sum_{\alpha} Q_j^{\alpha} + \sum_{\alpha} R_j^{\alpha} \quad \text{Eq. 3-17}$$

- The Darcy flux (**q**) can be described in terms of the phase velocity and porosity (ϕ):

$$\boldsymbol{q}^{A} = \boldsymbol{u}^{A}\phi \quad \text{Eq. 3-18}$$

- The mass of the aqueous phase is described in Eq. 3-2.

- The aqueous concentration $C^A$ is calculated for each species by distributing the species mass across the pore volume of each grid cell. The resulting aqueous concentration is then compared to the solubility limit $C^*$.
    - If the solubility is *below* the limit, the concentration remains unchanged.
    - If the solubility is *above* the limit, the aqueous concentration is set to the solubility limit and the excess amount of mass is assigned to the precipitated mass. It is not clear if the precipitated molar volume used in UFD decay is handled analogously in NWT for the solid volume fraction of the pore space. This would have to be examined further.
    - The sorbed mass is calculated by multiplying the aqueous concentration by the elemental $K_d$ value.
    - If the solubility is *at* the limit before the sorbed mass is calculated, the aqueous concentration is reduced slightly below the limit by the sorbed mass.
- The precipitated and sorbed phases are immobile, so their masses can be described by the solid pore volume fraction $S_p$ or as a function of the aqueous concentration as described previously in Eq. 3-3 and Eq. 3-4.
- How are sources and sinks treated in NWT mode:
    - NWT uses source/sink terms for both the decay/ingrowth within the phase ($Q$) and the contribution from reactions ($R$).
    - The influx source/sink term is described in terms of the volumetric Darcy flux ($\mathbf{Q}$) or volumetric flow ($\boldsymbol{U}$) and volume ($V$).

$$Q_j^\alpha = \frac{\boldsymbol{U}_j^A}{V} M_j^A = S_l C_j^A \left( \boldsymbol{Q}_j^A / V \right) \quad \text{Eq. 3-19}$$

    - Similar to UFD Decay, the decay/reaction source term is defined by the Bateman equations in NWT. However, there is no treatment of branching ratios like in UFD Decay.

$$R_j^\alpha = \frac{\partial M_j^\alpha}{\partial t} = -\lambda_j M_j^\alpha + \lambda_{j-1} M_{j-1}^\alpha \quad \text{Eq. 3-20}$$

Equation Eq. 3-17 is discretized via integrated finite volume discretization and the implicit time discretization method. The residual $\mathcal{R}$ for phase α, species j, and time step $t$ is described as follows:

$$\mathcal{R}\left(M_j^{\alpha,t+1}\right) = G\left(M_j^{\alpha,t+1}\right) + F\left(M_j^{\alpha,t+1}\right) = 0 \quad \text{Eq. 3-21}$$

Here, $F$ represents the flux terms containing the Darcy fluxes, dispersion terms, concentration gradients, and interfacial areas. $G$ contains the non-flux terms including the mass of phases and influx/decay source/sink terms. When solving equation Eq. 3-21 as a linear system of equations with Newton-Raphson iterations $k$ and Newton step δ, the Jacobian relationship is defined as:

$$\sum_{j=1}^{n} \frac{\partial \mathcal{R}_i\left(M_i^{\alpha,t+1}\right)^k}{\partial M_j^\alpha} \left( \delta M_j^{\alpha,t+1} \right) = -\mathcal{R}\left(M_j^{\alpha,t+1}\right)^k \quad \text{Eq. 3-22}$$

When solving through Newton's method, the solution is updated at each iteration according to

$$\left(M_j^{\alpha,t+1}\right)^{k+1} = \left(M_j^{\alpha,t+1}\right)^{k} + \left(\delta M_j^{\alpha,t+1}\right) \quad \text{Eq. 3-23}$$

and the updated solution is used to evaluate the residual. Convergence of $\mathcal{R}$ is based on criteria using tolerances of $\delta M$ for the species and phase.

NWT mode contains the analytical robustness needed to track the masses of phases used in UFD Decay. However, the methodology used to determine solutions for the various phases is different, and therefore, a full one-to-one mirroring of results for a given problem setup should not be expected.

### 3.2.5 Future Proposed Modifications to NWT Mode

1. ***Include branching ratios in radioactive decay in NWT mode:*** The radioactive reaction object in NWT (radioactive_decay_rxn_type) could be refactored to include isotope branching ratios so that Eq. 3-20 will resemble Eq. 3-11. While the RADIOACTIVE_DECAY keyword syntax can be maintained, an alternate input syntax using ISOTOPE keyword and sub-blocks can be added as an alternative. The UFD Decay properties listed under ELEMENT can similarly be listed under NWT's SPECIES keyword.
2. ***Allow $K_d$ functions in NWT mode***: The solubility variable element_type in UFD Decay is an analog to the solubility limit variable species_type in NWT. However, NWT would need an expanded object to handle $K_d$ variation, since the NWT currently implements only constant $K_d$. Features to include are the ability to (a) handle a $K_d$ dataset (specified with DATASET keyword in UFD Decay) and (b) to allow the smectite-to-illite clay model to modify $K_d$. These features can be ported from UFD Decay element_kd_type object. However, since the sorbed concentration calculation in NWT is somewhat different compared to UFD Decay, these features will be added only after understanding these differences in baseline testing with static $K_d$.

## 3.3 Nuclear Waste Transport Mode Phase Partitioning

### 3.3.1 Motivation for NWT Update

The motivation behind updating nuclear waste transport (NWT) mode arose from several large-scale repository systems analysis simulations that experienced complete dry-out. The original formulation of phase partitioning had flaws in its implementation that was vulnerable to sorption causing mass balance errors.

### 3.3.2 Improved Phase Partitioning Based on Bulk Concentration

In FY24, phase partitioning in NWT mode was adjusted from its previous form. The pseudo-code below describes the original formulation:

```
if (water_saturation > 0):
    aqueous_concentration = bulk_concentration / (porosity * water_saturation)
    if (aqueous_concentration > solubility):
        mass_above_solubility = aqueous_concentration – solubility
        aqueous_concentration = solubility
        precipitated_mass = mass_above_solubility * water_saturation * porosity
    else:
        precipitated_mass = 0.0
    endif
```

```
    sorbed_mass = aqueous_concentration * Kd
        aqueous_mass = (aqueous_concentration  * water_saturation * porosity) – sorbed_mass
        aqueous_conc = aqueous_mass / (porosity * water_saturation)
endif
```

Here, the aqueous concentration was found from the total bulk concentration. Then, if the aqueous concentration was above solubility, the precipitated mass was found by subtracting the concentration at solubility from the aqueous concentration and setting the aqueous concentration to solubility. The sorbed mass was calculated by multiplying the aqueous concentration by the distribution coefficient $K_d$ value (m$^3$-water/m$^3$-bulk), and then that quantity was subtracted from the aqueous mass.

This formulation worked well for aqueous radionuclides that precipitated but was not sufficiently tested for radionuclides that sorb. In the case of sorption and precipitation, the sorbing phase of the component would occasionally reduce the aqueous concentration below solubility, causing mass balance errors.

The new formulation in NWT is based on the following equation:

$$C_{tot} = C_{aq}\phi S_w + C_{aq}K_d + C_{precip} \quad \text{Eq. 3-1}$$

where $C_{tot}$ (mol/m$^3$-bulk) is the total bulk concentration, $C_{aq}$ (mol/m$^3$-liquid) is the aqueous component concentration, $\phi$ (m$^3$-void/m$^3$-bulk) is the porosity, $S_w$(m$^3$-water/m$^3$-void) is the water saturation, $K_d$ (m$^3$-water/m$^3$-bulk) is the linear sorption coefficient, and $C_{precip}$(mol/m$^3$-bulk) is the bulk concentration of the precipitated solid phase. The new implementation in PFLOTRAN is presented in the pseudo-code below:

```
aqueous_concentration = min(solubility, total_bulk_conc / (porosity * saturation * Kd)
aqueous_mass = aqueous_concentration * (water_saturation * porosity)
sorbed_mass = aqueous_concentration * Kd
precipitated_mass = max(0.0, total_bulk_conc – aqueous_mass – sorbed_mass)
```

Here, the sorbed and aqueous phases are computed first as the minimum of solubility, or

$$C_{aq} = \frac{C_{tot}}{\phi S_w + K_d}. \quad \text{Eq. 3-2}$$

The sorbed concentration is calculated as $C_{sorb} = C_{aq}K_d$. The precipitated concentration is then found by subtracting the sorbed and aqueous phase concentrations from the total bulk concentration.

## 3.4 Miscellaneous Improvements, Regression Tests, and Debugging of NWT Mode

### 3.4.1 Concentration Input/Output by Regions in NWT Mode

The GIRT process model allows for output of component masses within a specific grid cell, and across an entire region. NWT mode did output the total global bulk mass concentration but not by region. NWT mode was updated to allow concentrations to be defined by regions, and to output the total bulk, aqueous, sorbed, and precipitated/mineral concentrations by regions

### 3.4.2 NWT Mode Across Flow Modes

PFLOTRAN version 5.0 has 16 regression tests using NWT mode, 6 of which use the WIPP flow mode. Thus, not all 4 flow modes have equal coverage. To monitor NWT mode integrity as developments were undertaken, new regression tests were created employing NWT mode coupled with GENERAL, RICHARDS, WIPP, and TH (thermal-hydrologic) flow modes with unified domain characteristics. The tests involve a 100 × 50 × 100 m (*x,y,z*) domain (Figure 3-5), where an observation point at the center monitors the concentration of $^{241}$Am ($t_{1/2}$ = 432 y). $^{241}$Am is defined with a Dirichlet zero-gradient transport condition of $1\times10^{-1}$ mol/m$^3$ liquid at the surface. The $^{241}$Am radionuclide has a solubility $3.085\times10^{-3}$ mol/m$^3$-liquid and precipitate molar density of 3.861 mol/m$^3$mineral. The sorption distribution coefficient ($K_d$) is set to zero to allow $^{241}$Am to move at the fluid velocity determined by the flow model.

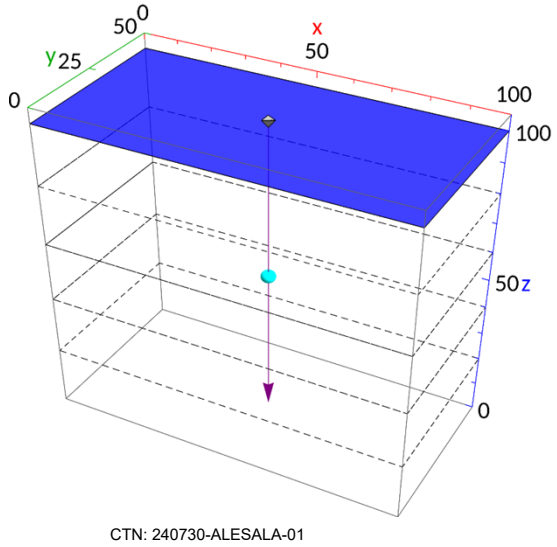



**Figure 3-5. Diagram of 100 × 50 × 100-meter (*x,y,z*) domain for NWT mode test applied to four PFLOTRAN flow modes; top blue surface is the recharge boundary with 5.0 MPa boundary condition and the blue center point is the $^{241}$Am concentration observation point; z is direction of gravity.**

The material properties of this domain are described in Table 3-4, where a Second-Modified Brooks-Corey characteristic curve (i.e., BRAGFLO KRP=4 option) is used for the mostly saturated system. Liquid residual saturation is set to zero while gas residual saturation is set to 10%. The tests apply a liquid flux (recharge) of 1 m/yr (including a zero-gas flux for GENERAL mode) and a hydrostatic pressure gradient with 5.0 MPa boundary condition at the top surface.

**Table 3-4. Properties of porous medium used in NWT mode tests.**

| Symbol | Keyword | Value | Units |
|---|---|---|---|
| | CHARACTERISTIC_CURVES | KRP4 (BRAGFLO) | n.a |
| | LIQUID_RESIDUAL_SATURATION | 0.0 | - |
| | GAS_RESIDUAL_SATURATION | 0.1 | - |
| | TOLC | 0.5 | - |
| | LIQUID_RESIDUAL_SATURATION | 0.0 | - |
| | GAS_RESIDUAL_SATURATION | 0.1 | - |
| | TOLC | 0.5 | - |

| Symbol | Keyword | Value | Units |
|---|---|---|---|
| $\varphi$ | POROSITY | 0.3 | - |
| | SOIL_COMPRESSIBILITY_FUNCTION | POROSITY_EXPONENTIAL | n.a |
| $\alpha$ | POROSITY_COMPRESSIBILITY | $10^{-9}$ | - |
| $P_{ref}$ | SOIL_REFERENCE_PRESSURE | INITIAL_PRESSURE | Pa |
| $c_p$ | HEAT_CAPACITY | 830 J/kg-C | |
| $\kappa_T(S_l)$ | THERMAL_CONDUCTIVITY_FUNCTION | DEFAULT | n.a |
| $\kappa_T^{DRY}$ | THERMAL_CONDUCTIVITY_DRY | 5.5 W/m-C | W/m-C |
| $\kappa_T^{WET}$ | THERMAL_CONDUCTIVITY_WET | 7.0 W/m-C | W/m-C |
| $\rho$ | ROCK_DENSITY | 2650 kg/m$^3$ | kg/m$^3$ |
| $K_\perp$ | PERM_ISO | $10^{-12}$ m$^2$ | m$^2$ |

Figure 3-6 shows that the total bulk concentration of $^{241}$Am at the observation point decreases over time. The four different flow modes show excellent agreement given common transport conditions with NWT. The liquid pressure behavior is shown in Figure 3-7, where the pressure rises monotonically from the initial hydrostatic pressure of 2.5 kPa. Differences in behavior are due to differences in problem initialization and the different theoretical frameworks for the flow modes. For example, the WIPP problem is defined with a saturation value slightly less than one. Also, RICHARDS and WIPP modes are isothermal, while both GENERAL and TH mode use Dirichlet temperature conditions at 20°C. Therefore, while flow characteristics behave differently, the transport characteristics are shown to be the same.

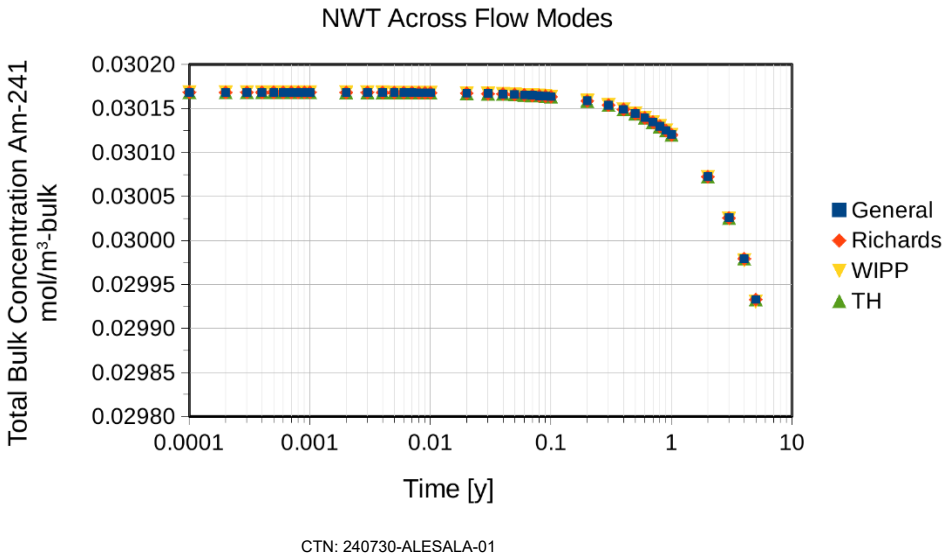



**Figure 3-6. Total bulk concentration of $^{241}$Am over time for NWT mode with the four different flow modes.**

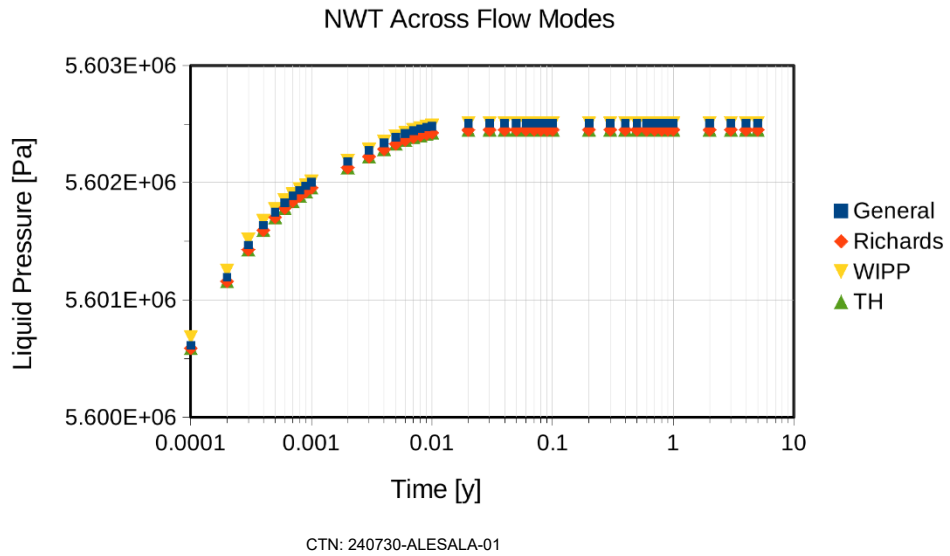



**Figure 3-7. Liquid pressure over time for the four flow modes in the NWT test domain.**

### 3.4.3 Debugging NWT Mode

The waste form process model computes, based on an initial inventory, radionuclide decay and ingrowth within a waste form, the canister degradation, instant release of radionuclides upon canister breach, and dissolution into the pore water (Frederick et al., 2016). The waste form process model is computed sequentially, following the reactive transport process model (GIRT or NWT). The instant release molality and instantaneous dissolution molality are computed based on the instant release fraction specified in the input deck, the waste form volume and density, and the volume of pore water. The instant release fraction (mol-radionuclide/kg water) is then added to the solution vector. Because the GIRT and NWT process models use different primary variables (mol/kg and mol/m$^3$-bulk, respectively), the addition of radionuclides to the reactive transport process model solution vectors resulted in an error of several orders of magnitude. This error was solved by adding a case statement that checks the process model and uses the corrects units for GIRT or NWT mode.

Upon switching from GIRT to NWT, the concentrations at various observation points were compared between the two process models to confirm their conformance with the waste form process model (Figure 3-8). Through iterative code improvements, and close work with the modeling team, the solution provided by NWT compared with GIRT improved. The percent difference between the two codes was typically less than 1%.

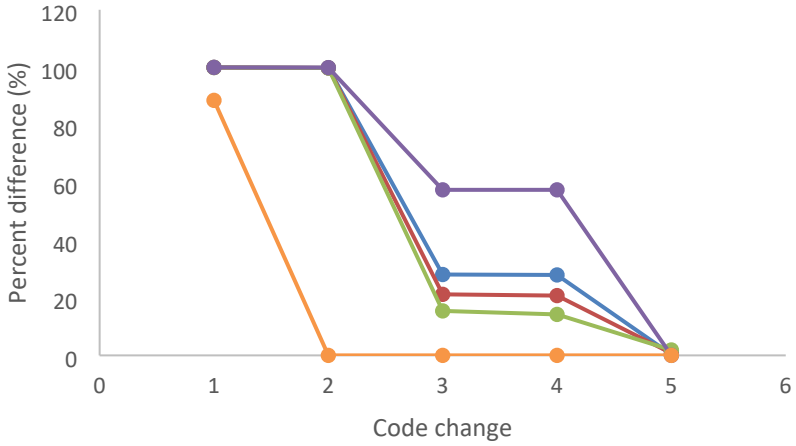



**Figure 3-8. Agreement improved between GIRT and NWT modes with bug fixes. The percent difference from the GIRT baseline in $^{129}I$ concentration at 5 observation points are plotted, with the two process models converging as sequential changes to NWT mode were made.**

## 3.5 Quasi Implicit Wellbore Model

We have developed a quasi-implicit wellbore process model within PFLOTRAN to simulate how a drilling intrusion could impact flow and transport in a deep geologic repository system. This capability was initially developed by the WIPP performance assessment project. By employing a quasi-implicit approach, this allows for a balance between computational efficiency and high-fidelity results.

The quasi-implicit wellbore process model uses Darcy flow principles within the wellbore, while calculating the one-dimensional wellbore grid separately from the three-dimensional repository system model. This approach eliminates the need for mesh refinement around the wellbore, which would be needed to model the wellbore explicitly in the three-dimensional model domain. By reducing the number of mesh elements required, this significantly decreases computation time and enhances overall computational efficiency. In addition, a wellbore model can be easily added to an existing model.

The quasi-implicit wellbore process model incorporates casing and uncasing indices for the wellbore and uses Peaceman's formula (Peaceman, 1978) to connect the wellbore to the reservoir. This represents flow dynamics between the well and the surrounding geological formations. The wellbore model functions as a subprocess of the PFLOTRAN flow mode used for the repository simulation. Transport within the wellbore is a subprocess of the PFLOTRAN transport mode used for the repository simulation. This method ensures seamless sequentially coupled integration and interaction between the different model components.

The implementation of the quasi-implicit wellbore process model in PFLOTRAN adds capability to GDSA Framework to consider human intrusion (drilling intrusion) scenarios. Compared to explicit modeling of wellbore in a 3D domain, it improves the precision of simulations, enabling better prediction of the behavior of radionuclides in the event of a wellbore intrusion.

## 3.6 Implicit Salt Transport Option

### 3.6.1 Overview

Fully implicit solute transport was added to GENERAL mode process model of PFLOTRAN in FY22 (i.e., the GENERAL process model now includes a solute/salt mass conservation equation in addition to the conservation equations for water, air, and energy—Nole et al., 2022). The addition of this conservation equation allows for implicit solute transport. Unconditionally stable solute transport is of particular importance because of the strong dependence of density on salinity. This process model has been continually improved since its original implementation, with new capability added in FY24 for specifying both soluble and insoluble rocks within a simulation domain, options to improve simulation convergence, and bug fixes in support of larger repository simulations.

Prior to the development of Solute Salt option, the standard reactive transport process model in PFLOTRAN, GIRT, would be sequentially coupled to the flow solution. The sequential coupling presents several issues, particularly related to the geologic disposal of radioactive waste. First, the nearfield regions of radioactive waste repositories can often reach temperatures above the water boiling point, which eventually dry out the system and cause solutes to precipitate in the solid phase. Second, GIRT requires the presence of a liquid phase because the primary species are defined as aqueous components. Finally, radioactive waste disposal in salt incorporates many physical processes impacting each other that should be solved implicitly. For example, as a radioactive waste repository in salt heats up, vapor and brine transport alter the pore water content, which causes salt to dissolve in some regions and precipitate in others. These changes in porosity alter the permeability and thermal conductivity and influence the phase pressures as the pore space closes. Also, the dissolution of the solid phase into the liquid phase may sufficiently change the liquid density to promote density driven flow (as described later in Section 3.6.3)

The conservation equations for GENERAL mode have been described in previous PFLOTRAN development reports (i.e., Nole et al., 2022 and Nole et al., 2023) and are also summarized here. GENERAL mode solves conservation equations for two or three mass components (air, water, and optionally salt), and energy. The mass conservation equations for water and air take the following form:

$$\frac{\partial}{\partial t}\varphi\left(s_l\rho_l x_i^l + s_g\rho_g x_i^g\right) + \nabla\cdot\left(\boldsymbol{q}_l\rho_l x_i^l + \boldsymbol{q}_g\rho_g x_i^g - \varphi s_l D_l\rho_l\nabla x_i^l - \varphi s_g D_g\rho_g\nabla x_i^g\right) = Q_i. \qquad \text{Eq. 3-3}$$

Here, the component mole fraction is $x_i^{l,g}$, the phase saturation is $s_{l,g}$, the phase density is $\rho_{l,g}$, the porosity is $\varphi$, time is $t$, the Darcy flux of each phase is $\boldsymbol{q}_{l,g}$, diffusion coefficient is $D_{l,g}$, and the source/sink term is $Q_i$.

Salt exists only as a dissolved component, or as a precipitated solid (not gaseous form). As a result, the mass conservation equation is slightly different when the Solute Salt option card is used:

$$\frac{\partial}{\partial t}\varphi\left(s_l\rho_l x_{NaCl}^l + s_p\rho_p x_{\mathrm{NaCl}}^p\right) + \nabla\cdot\left(\boldsymbol{q}_l\rho_l x_{\mathrm{NaCl}}^l - \varphi s_l D_l\rho_l\nabla x_{\mathrm{NaCl}}^l\right) = Q_{\mathrm{NaCl}} \qquad \text{Eq. 3-4}$$

The variables of Eq. 3-4 are the same as in Eq. 3-3, but the phase saturation of the precipitated phase is also considered in the accumulation term (the maximum dissolved salt in the liquid phase is itself a function of pressure and temperature). As the precipitated phase is immobile, and salt is not considered in the gas phase, the flux of salt depends on the advection and diffusion in the liquid phase.

In a matrix comprised entirely of soluble salt (such that the rate of dissolution and precipitation is high enough that the solid phase is always in equilibrium with the liquid phase), the porosity can change based on liquid and vapor advection, as well as changes in solubility arising from changing temperature. Rather than considering only a precipitated phase in the pore space as in Eq. 3-4, the entire rock matrix is conserved within the salt mass conservation equation:

$$\frac{\partial}{\partial t}\left(\varphi\left(s_l\rho_l x^l_{\mathrm{NaCl}}\right)+(1-\varphi)\rho_r\right)+\nabla\cdot\left(\boldsymbol{q}_l\rho_l x^l_{\mathrm{NaCl}}-\varphi s_l D_l\rho_l\nabla x^l_{\mathrm{NaCl}}\right)=Q_{\mathrm{NaCl}} \quad \text{Eq. 3-5}$$

The terms here are the same as in Eq. 3-4 but include the mass of the rock matrix in the accumulation term.

The energy conservation equation remains the same and is summed over the present phases.

$$\sum_{j=l,g,p}\left\{\frac{\partial}{\partial t}\left(\varphi s_j\rho_j U_j\right)+\nabla\cdot\left(\boldsymbol{q}_j\rho_j H_j\right)\right\}+\frac{\partial}{\partial t}\left((1-\varphi)\rho_r c_p T\right)-\nabla\cdot\left(\kappa\nabla T\right)=Q \quad \text{Eq. 3-6}$$

where the variables are the same as in Eq. 3-3. Additionally, $U_j$ is the phase internal energy, $H_j$ $is$ the phase enthalpy, $c_p$ is the heat capacity of the rock, $\kappa$ is the thermal conductivity, and $T$ is temperature.

### 3.6.2 Defining both Soluble and Insoluble Materials

In FY24, the capability to define both soluble and insoluble materials was added to the Solute Salt option in the GENERAL mode (i.e., materials that have a soluble rock matrix are defined using the SOLUTE SALT option within the GENERAL mode OPTIONS block, as shown below). This capability is useful for simulating other, insoluble geologic and engineered materials adjacent to the salt host rock in the repository.

```
SIMULATION
 SIMULATION_TYPE SUBSURFACE
 PROCESS_MODELS
  SUBSURFACE_FLOW flow
   MODE GENERAL
   OPTIONS
    SOLUTE SALT
    SOLUBLE_MATERIALS soil1
   /
  /
 /
END
```

The SOLUBLE_MATRIX keyword has been removed. Materials listed after the SOLUBLE_MATERIALS keyword will use Eq. 3-5, for salt mass conservation, and otherwise will use Eq. 3-4 by default.

The FLOW_CONDITION blocks have also changed slightly. Porosity is now defined within the material properties (i.e., porosity is no longer defined as a flow condition). Rather, the AT_SOLUBILITY type is used as a flow condition for SALT_MOLE_FRACTION in both soluble and insoluble rock types. A sample FLOW_CONDITION block is below (however, the numerical value assigned to SALT_MOLE_FRACTION is not used):

```
FLOW_CONDITION initial
 TYPE
  LIQUID_PRESSURE DIRICHLET
  MOLE_FRACTION DIRICHLET
  SALT_MOLE_FRACTION AT_SOLUBILITY
  TEMPERATURE DIRICHLET
 /
 LIQUID_PRESSURE 1.5d6
```

```
  MOLE_FRACTION 1.d-8
  SALT_MOLE_FRACTION 1.D0
  TEMPERATURE 25.D0
/
```

### 3.6.3 Example of Halite Dissolution and Subsequent Density Driven Flow

An example of halite dissolution and subsequent density driven (free convection) flow is displayed in Figure 3-9. Here, a halite region is initialized in the upper-left corner, from (*x*,*z*) coordinates of (0,19) to (15,20). The salt mole fraction in the halite region is at solubility, while the salinity of the background region was set at a low value ($10^{-8}$). As the halite dissolves, the denser, saline fluid moves into the domain.

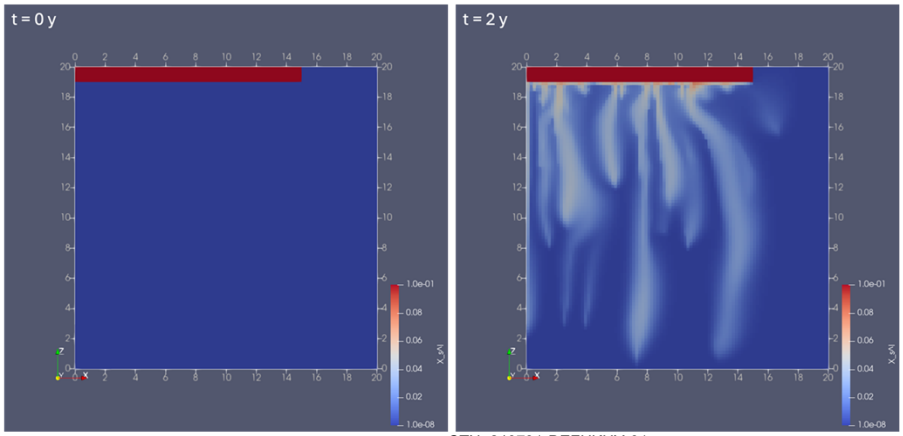



**Figure 3-9. Halite dissolution and subsequent density-driven free convection flow.**

As the halite dissolves, the porosity increases from 0.15 to 0.235 in the lower-right corner of the halite region after 2 years (Figure 3-10).

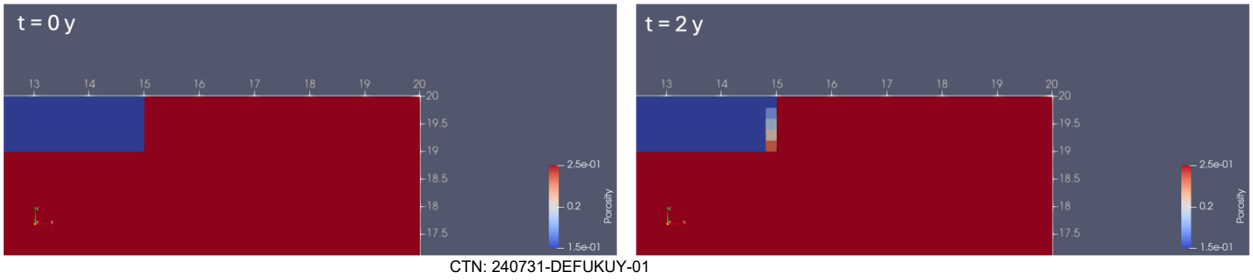



**Figure 3-10. Porosity change in the halite rock with dissolution.**

# 4. PROCESS MODEL SUPPORT

## 4.1 Dual Continuum

### 4.1.1 Code Updates

As noted in FY23 (Nole et al., 2023), matrix diffusion coupled with sorption is considered one of the most important radionuclide retardation mechanisms in fractured crystalline host rocks (Neretnieks et al., 2017). One way of modeling diffusive transport between a rock matrix and fractures is by representing the phenomenon as a Fickian diffusion process over a dual porosity (dual continuum) system. The dual continuum model in PFLOTRAN models a secondary continuum (matrix) coupled perpendicular to the primary continuum (fracture) modeled as a disconnected one-dimensional domain that is referred to as the DCDM (Dual Continuum Disconnected Matrix) model. Advective and diffusive transport can occur in the primary continuum, but only one-dimensional diffusive transport can occur in the secondary continuum. The secondary cells cannot interact with secondary cells associated with other primary cells. The equations for the primary and secondary continua are solved separately and coupled together by a mass exchange flux term assuming symmetry along the axis dividing them. Since secondary continua are isolated from one another, PFLOTRAN's multiple continuum model is ideal for complex full-scale crystalline repositories using high performance computing. The model implementation is described in more detail in Nole et al. (2021).

The dual continuum model in PFLOTRAN has undergone several developments in FY24. The ability to have a variable number of secondary cells within the multiple continuum block has been added. This ability is useful when simulating a model with a heterogenous effective diffusion coefficient or the multiple continuum model is turned off in a certain area of the domain (such as in the repository for a spent nuclear fuel reference case). A variable number of secondary cells can be input as a dataset or by specifying different values in multiple material blocks.

Additionally, a bug was identified in the log formulation in the multiple continuum model when using two phases. The indexing for the partial derivative of the total concentration in the gas phase divided by the concentration in the aqueous phase, which is used in the Jacobian calculation, was incorrect and fixed. The derivative is now correctly indexed by each component in the gas phase.

### 4.1.2 Using UFD Decay Mode and Dual Continuum in DECOVALEX Task F1

The Dual continuum model in PFLOTRAN was applied in the analysis of a generic spent nuclear fuel repository in fractured crystalline rock as described in the international research project DECOVALEX Task F1 (LaForce et al., 2023). The reference case assumes isothermal conditions, steady state flow, and two different radionuclide transport scenarios. In FY24, the dual continuum model was successfully used to model the full radionuclide inventory from Task F1 (LaForce et al., 2023). The radionuclide inventory contains $^{129}$I and a $^{238}$U decay chain. One waste package fails at the beginning of the simulation and the rest fail at 50,000 years.

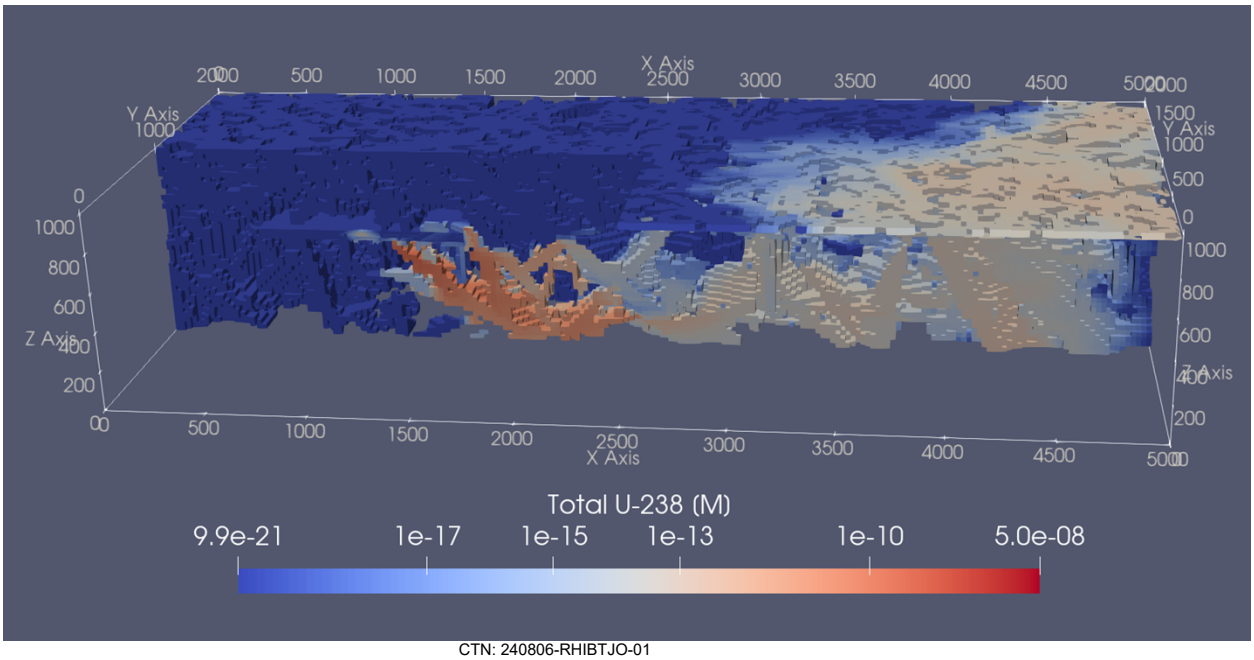
In the analysis, each primary cell had a 12.5 cm secondary continuum which was discretized with 100 secondary cells. The simulation used 256 cores and took ~60 minutes to reach 100,000 years.

The model solution was sensitive to the truncate concentration specified in the CHEMISTRY block. For the simulation to go beyond the first timestep, the truncate concentration had to be set an order of magnitude smaller than used when modeling a conservative tracer in the reference case. An error was also identified and fixed in the allocation of the sorbed tracer amount when using the UFD Decay process model in the secondary continuum.

The concentration of $^{238}U$ in the DCDM domain is plotted at 40,000 and 100,000 years in Figure 4-1 and Figure 4-2, respectively. $^{238}U$ is included to appropriately simulate uranium solubility (the isotope partitioning model requires inclusion of all dominant isotopes) and to observe $^{238}U$ decay into $^{234}U$. Only the cells containing fractures are used in the dual continuum model; the dual continuum model is turned off inside the repository, which is a feature that was developed in FY23.

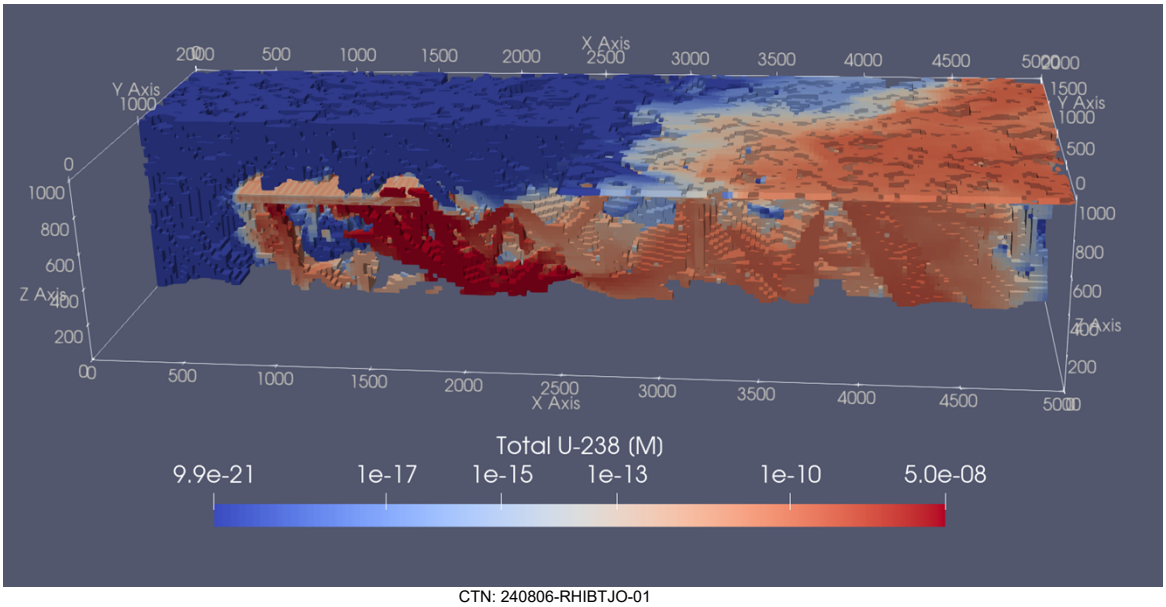
The mean of ten fracture realizations and the 95% confident interval of the mean of the cumulative flow and mass flow of each radionuclide out of the top surface of the domain on the low point ($3700 < x < 5000$ m) is plotted in Figure 4-3. The cumulative flow and mass flow calculated in the DCDM are slightly higher than values calculated with the Equivalent Continuous Porous Medium (ECPM). Possibly, numerical dispersion is less with DCDM than ECPM because diffusion into the matrix occurs over 12.5 cm in DCDM versus 25 meters in the ECPM.

Depending on the number of cells used in the secondary continuum and the density of the fracture network, the dual continuum model can also decrease run time when compared to the ECPM reference case. Figure 4-4 shows the fluxes across the top surface out of the low point of the domain for three different effective diffusion coefficients. The effective diffusion coefficient specified in F1 task specification is $10^{-13}$ $m^2/s$. If the diffusion coefficient is increased to $10^{-12}$ $m^2/s$, the cumulative $^{129}I$ exiting the domain over the low point is decreased by ~1 mole. If the diffusion coefficient is increased another order of magnitude to $10^{-11}$ $m^2/s$, the cumulative $^{129}I$ exiting the domain over the low point is decreased by only another ~0.2 moles.



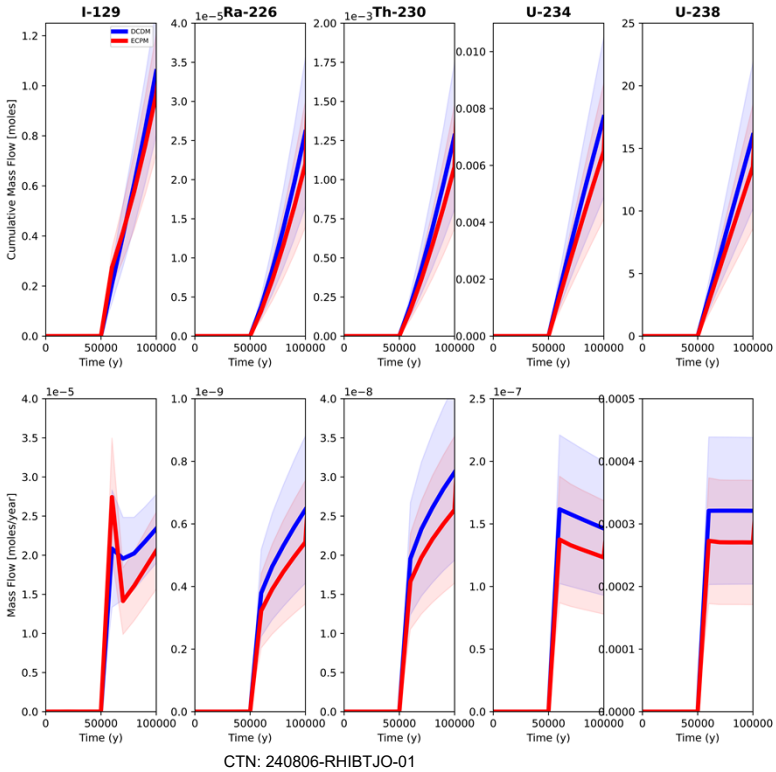

CTN: 240806-RHIBTJO-01

**Figure 4-1. $^{238}U$ concentration in the primary continuum on a slice of the DECOVALEX domain at 40,000 years, one waste package has breached.**




CTN: 240806-RHIBTJO-01

**Figure 4-2. $^{238}$U concentration in the primary continuum on a slice of the DECOVALEX domain at 100,000 years, all waste packages have breached.**




CTN: 240806-RHIBTJO-01

**Figure 4-3. Mean of 10 fracture realizations and 95% confidence interval of the cumulative mass and mass flow of five radionuclides for DCDM (blue) and ECPM (red) transport models out of the top DECOVALEX domain on the low point (3700 < x < 5000 m).**

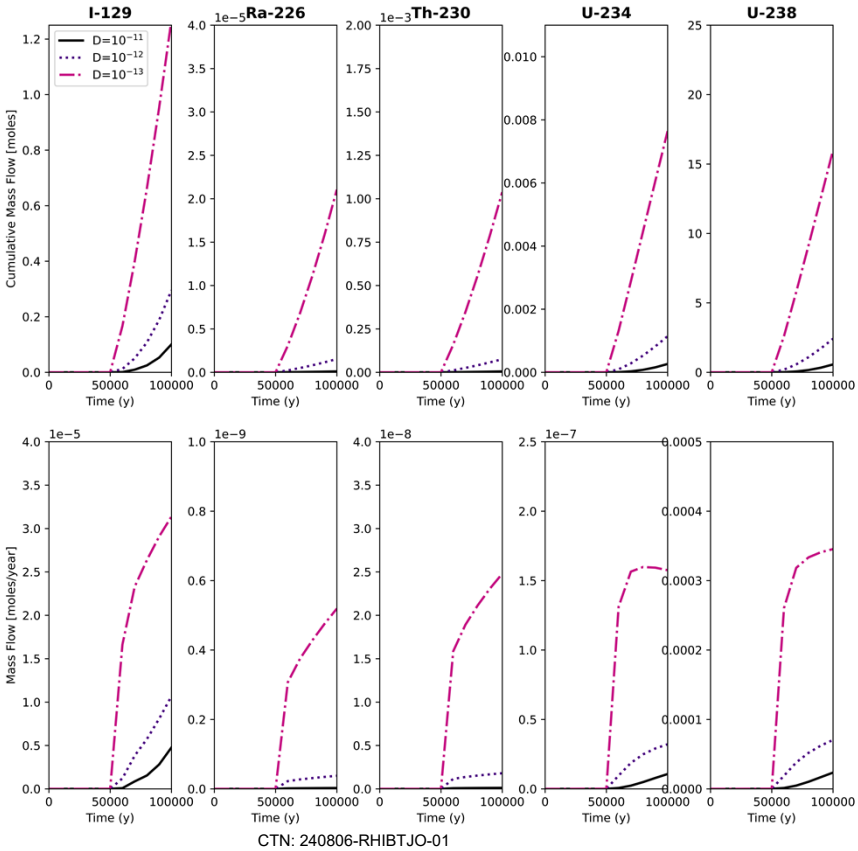
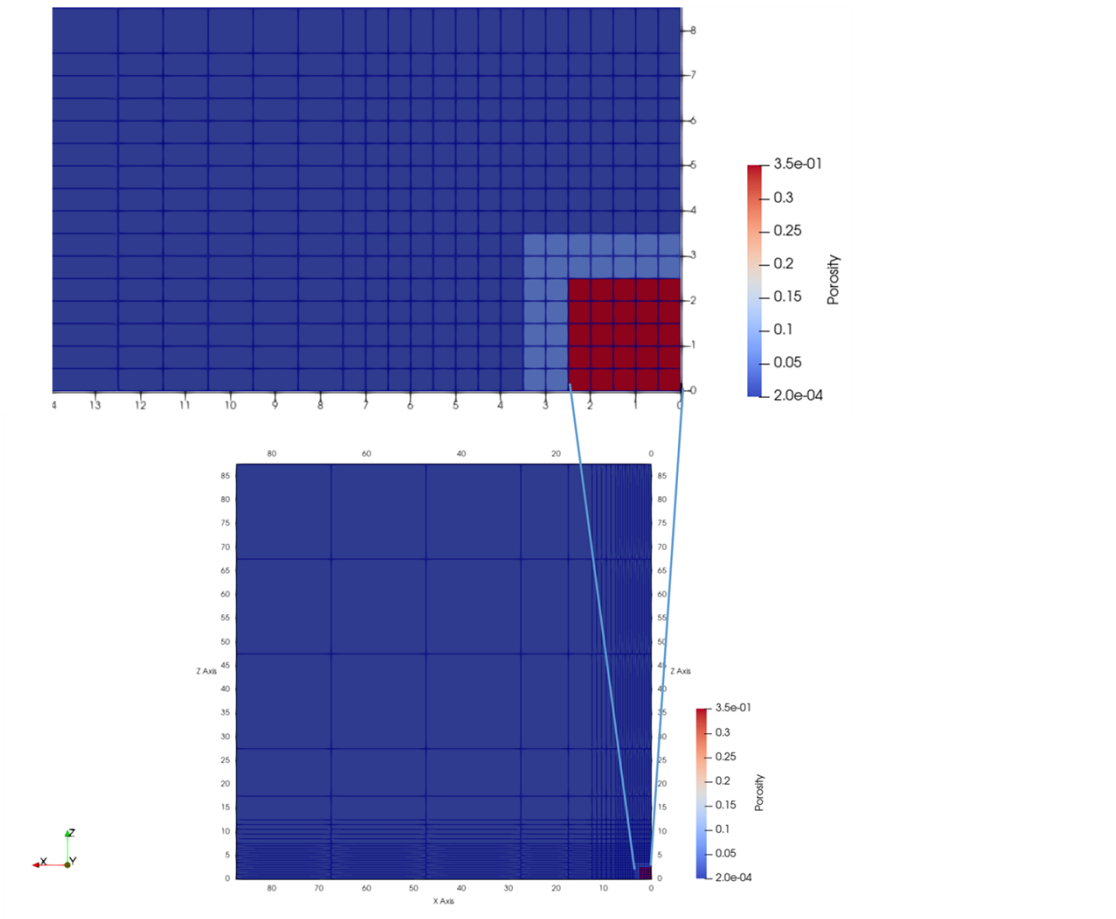



**Figure 4-4. Fluxes across the low point for one realization simulated with the DCDM with 3 different effective diffusion coefficients compared to the base case specified in the DECOVALEX F1 task ($10^{-13}$ $m^2/s$).**

## 4.2 Modeling Heat Flux and Fluid Flow in Creeping Salt

In conjunction with international collaboration on salt with Germany (RANGERS Sandia/BGE technical exchange collaborative project—Mills et al., 2024), a quarter-symmetry, single drift problem was developed to model heat flux and fluid flow in a system undergoing creep closure to test the new PFLOTRAN capabilities implemented in GENERAL mode, as described in Section 3.6. The system used a structured mesh, more finely discretized near the waste package, and more coarsely meshed in the far-field (Figure 4-5). Here, three regions are defined with different material properties: crushed salt, excavation damaged zone (EDZ), and intact salt. Waste packages were not individually modelled; instead, the heat source was distributed uniformly across the crushed salt. The three regions have varying material properties, described in Table 4-1.




CTN: 240731-DEFUKUY-02

**Figure 4-5. Schematic of the quarter-symmetry drift approximating creep closure of salt.**

**Table 4-1. Material properties of the three regions.**

| Material | Porosity | Tortuosity | Rock density ($kg/m^3$) | Permeability ($m^2$) | Heat capacity (J/kg-K) |
|---|---|---|---|---|---|
| Crushed salt | 0.35 | 1.5 | 1820 | $8.79\times10^{-15}$ | 562 |
| EDZ | 0.1 | 1.0 | 2100 | $3.16\times10^{-16}$ | 850 |
| Intact salt | 0.0002 | 1.0 | 2200 | $1.00\times10^{-20}$ | 864 |

The effect of creep closure of the disposal room was approximated by reducing the porosity of the crushed salt and EDZ regions. The porosity was reduced by adding a solid mass of salt as a source to these regions. In the crushed salt region, the porosity was reduced from 35% to 3% over the first 100 years, and in the EDZ, the porosity was reduced from 10% to slightly under 5% over the same period. A heat source representing the heat from a waste package was uniformly applied to the crushed salt region. By controlling the mass of salt to represent creep closure, all non-porosity variables and several material properties respond to the imposed reduction in pore volume. There should be a dependence of closure rate on pore pressure and other variables not considered here (i.e., if pore pressure reaches lithostatic pressure, creep closure should stop). The dependence of creep closure on pore pressure, as well as coupling with geomechanics, will be developed in the future.

The evolution of the system is displayed in Figure 4-6. The average (solid lines) and bounds (shaded regions) of the drift (red), EDZ (green), and near-EDZ intact salt (blue) are displayed for gas pressure, gas density, liquid saturation, temperature, porosity, liquid pressure, permeability, and thermal conductivity. The porosity is controlled in the EDZ and the drift by adding a mass of salt to the region that corresponds to the average closure rate. However, because the heat source, fluid flow, and solubility in the drift changes (but not the EDZ), the porosity within the drift and the EDZ do not decrease uniformly. As the temperature increases in the drift, the liquid water begins to vaporize, then it advects to cooler regions where it recondenses (the corner of the drift and the EDZ). The condensation of water in the EDZ then partially dissolves the rock, which increases its porosity. The liquid and gas pressures and saturations are altered by both pore closure and temperature. As the system heats up and the pore space closes, the liquid and gas pressure increases. The liquid phase, being much less compressible than the gas phase, increases in saturation.

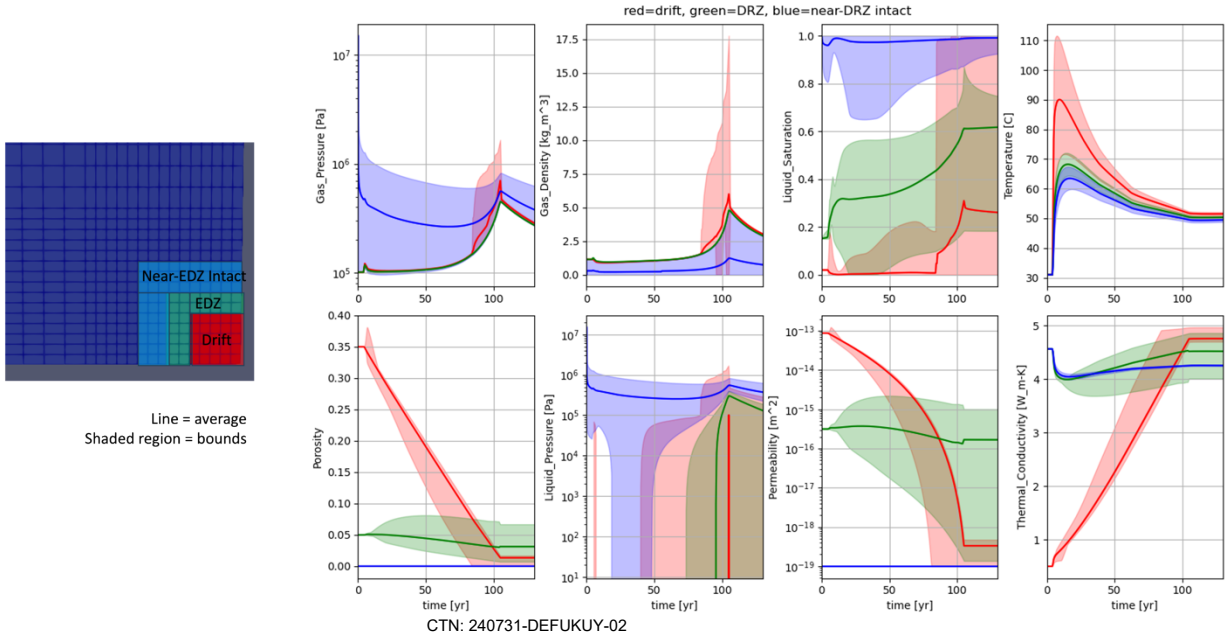



**Figure 4-6. Pressure, saturation, permeability, temperature, and thermal conductivity responses to creep closure.**

Thermal conductivity and permeability also change during creep closure. Intact salt has much higher initial thermal conductivity than granular salt. As the pore space closes, the reduction in porosity leads to an increase in thermal conductivity. The dependence of thermal conductivity on porosity was implemented using the LINEAR_RESISTIVITY thermal characteristic curve. Temperature also slightly reduces the thermal conductivity of salt, so the regions that do not experience as much closure see a reduction in thermal conductivity. Permeability here is solely a function of porosity, and exponentially decreases with porosity by a power-law relationship: $k = k_0\phi^n$, where $k$ is the permeability, $k_0$ is the initial intrinsic permeability, and n is a dimensionless exponential fitting parameter, defined in this system as 2.0.

The expansion of capability in modeling creep closure of salt has improved the conceptual understanding of fluid and heat transfer immediately post-closure. Future improvements can be made to simulate the feedback between creep closure and pore pressure, geomechanics, and mineral precipitation/dissolution.

## 4.3 General Debugging

Significant effort was spent this year on improving and debugging existing capability for use in large-scale models. Debugging is an aspect of code development that is largely invisible to users. A bug that causes the program to stall or crash can arise from a single line of code, typically written by a different developer than the one fixing it. While bug fixes can appear simple, even a small code change can represent many hours of work identifying the bug, fixing it for the identified case, and ensuring that the fix does not cause new problems elsewhere. By working closely with PFLOTRAN users, the SNL development team has improved how bugs are communicated from users to developers and has improved our internal capability of addressing code problems as they arise.

### 4.3.1 Perturbation Fix

The gas-precipitate state did not assign a perturbation to all the degrees of freedom, resulting in a not-a-number (NaN) value in the Jacobian. The phase change at the boundary resulted in a PETSc error. The perturbations are now set equal to the perturbations in the `GAS_STATE` with the addition of a perturbation for either porosity or precipitate saturation for the fourth degree of freedom.

### 4.3.2 Transient Condition Fix

The `SALT_MOLE_FRACTION` flow condition would accept a list of time-dependent concentrations but would maintain a constant boundary condition across time. The function `FlowConditionGeneralIsTransient()` was not testing `FlowSubConditionIsTransient (condition%salt_mole_fraction)`, for a transient condition, so the `FlowConditionGeneralIsTransient` Boolean was never set to *true* when only `SALT_MOLE_FRACTION` was transient. This was fixed by adding `FlowSubConditionIsTransient(condition%salt_mole_fraction)` to the conditional for GENERAL mode transient conditions.

### 4.3.3 Central Difference Jacobian

To improve convergence, an option for central difference Jacobians was implemented in `GENERAL` mode. Rather than forming the derivative based on a single perturbation, the central difference Jacobian perturbs the primary variables in both directions. While this change improves the accuracy of the derivative, it also doubles the number of computations required to form the Jacobian. A minimum central difference perturbation can be specified in the `GENERAL/OPTIONS` block, as `MIN_CENTRAL_DIFFERENCE_PERT`.

### 4.3.4 GENERAL Mode Options

`MIN_POROSITY` in the `GENERAL` flow mode provides a lower bound on the porosity for soluble material regions. This option is useful for simulations where the porosity reaches a low value, and fluid becomes immobile. Occasionally, porosity reductions close to zero causes convergence issues with the fluid pressures as the pore space closes and immobile fluids pressurize.

`MIN_LIQUID_SATURATION` sets a lower bound on the liquid saturation, preventing complete dry-out.

`SALT_SOURCE_MIN_POROSITY` turns off the salt source term after a specified porosity has been reached. In recent simulations, creep closure has been approximated by adding salt mass to the control volume as a source term. By setting this flag, the creep closure will stop once a minimum porosity has been reached.

`SALT_SOURCE_MAX_PRESSURE` sets an upper limit on the pore pressure as salt is being added to the system when approximating creep closure. After the maximum pressure has been reached, the source term will be set to zero.

### 4.3.5 Debugging Waste Form Process Model

A large-scale simulation of 7 million elements using 576 cores in parallel was stalling in the waste form process model of nuclear criticality in a dual-purpose canister (Price et al., 2022). Debugging larger problems running on a cluster such as these can be a daunting task, as it requires a different approach than smaller problems. Smaller problems, by comparison, can be easily visualized to identify errors and can be easily opened in a debugger. Because of the size of this simulation, it had to be executed on the cluster, which meant that the typical tools for debugging were not available. Attaching the debugger to a stalled program on the cluster is a potential route for debugging but can become unwieldy when trying to determine the part of the code each processor is trying to execute.

The first step in debugging this problem was to create a checkpoint file close to the simulation time where the program stalled. After creating a checkpoint file, a series of tests were performed to determine if the stall was related to the parallelism of the code. This was achieved by running on varying numbers of processors to see if it made any difference. Halving the number of cores (288) produced the same result: a stall in the waste form process model. Next, the program was run on the same number of nodes, but fewer cores (260): same result. To see if the error occurred in serial, the program was run on a single processor. The program did not stall in the waste form process model but triggered an error in a different process model. Finally, the program was run on one less core than the original simulation (575) which did not stall in the same place but stalled similarly about 100 timesteps later. These tests gave some indication that it might be related to the parallelism of this process model.

Talking with colleagues is another useful approach to debugging. These conversations revealed issues others had encountered before, specifically that using a different number of processors after a restart would lead to errors in the simulation. This situation indicated that the issue was related to the parallelism of the code. There were some previous concerns about domain decomposition when a waste package was assigned to elements that were then split between different processors. Based on this, a small test problem was developed to test this issue. The test problem was a 9-grid block domain with 3 waste packages, each covering a third of the domain. This way, one processor would do the calculations for all three waste packages, 2 processors would split the second waste package in half, 3 would split all three waste packages into their own processors, and 4 would split 2 and 3, but not 1 (Figure 4-7).

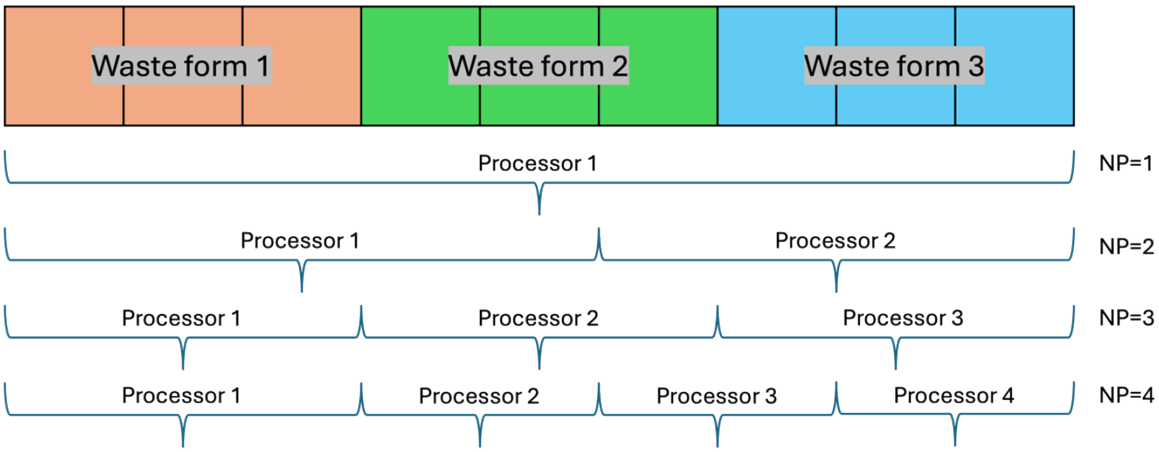



**Figure 4-7. Domain decomposition schematic for parallel testing of multiple waste forms split across processors.**

This test problem in conjunction with PFLOTRAN compiled with debugging flags (`-fcheck=all`, `-ffpe-trap=invalid`), we ran the test problem on a varying number of cores. The run-time flags revealed some

unallocated arrays unrelated to the problem we were experiencing, but after fixing those, the code produced a PETSc error in the waste form process model, where the bigger model was experiencing problems. The test problem was not stalling in the same way that the bigger model was, but it revealed an issue with the parallel nature of the code. By using the flag -start_in_debugger or -on_error_attach_debugger, PFLOTRAN would stop when it encountered an error, and print the stack trace of where the problem was occurring. The specific subroutine that was causing the error was CalcParallelSum, which took values from two processors, communicated them using MPI_Send and MPI_Recv, and returned a global sum. The values that were being sent through MPI_Send and MPI_Recv were the correct values and clicking through using the debugger and 4 processes did not show anything out of the ordinary.

By simply using print statements, the problem revealed itself. At various locations within the CalcParallelSum subroutine, print statements returned the rank, the location of the code, and relevant variables. The output on the terminal showed a rank of 15203582 that was responsible for performing a MPI_Send to rank 2. This was odd, because we were only using 4 processors, and we were not running this on over 15 million processors (nor do we have the capability to do so). The MPI rank was being corrupted by a different MPI process and was specifying an incorrect rank. This was solved by pointing the option object to this%realization%option, which provided the correct MPI ranks to be communicated during the CalcParallelSum calls.

Despite fixing the domain decomposition problem, the stall in the larger simulation remained. By using more print statements, a single rank was identified as waiting for the corresponding communication within the CalcParallelSum subroutine. By piping the terminal output into a text file and matching the MPI_Send/Recv ranks using a python script, the rank pair that caused the stall was identified. The subroutine responsible for the stall was PMWFInitializeTimestep, which used a conditional of: if (.not. Initialized(avg_sat_global)), and if (.not. Initialized(avg_temp_global)), to check whether an average temperature should be calculated across the gridblocks containing a waste form. This caused issues in parallel, as a rank containing multiple waste forms could have a temperature initialized by one waste form, but would not be initialized by the other waste form. This caused a mismatch in MPI_Send/Recv calls and a program stall. These conditionals were deemed unnecessary and removed, fixing the stall.

# 5. PERFORMANCE ADVANCEMENTS: ROADMAP FOR COMPUTING WITH GRAPHIC PROCESSING UNITS

## 5.1 Introduction

PFLOTRAN currently performs parallel computing through the multiple instruction, multiple data approach. Multiple CPUs are utilized through the message-passing interface (MPI) framework. This approach scales approximately with the number of nodes on high-performance computing clusters, though there is overhead in synchronizing the numerous processing threads and transferring data between nodes.

In comparison to CPUs, which perform a large range of operations over a single data point, Graphics Processing Units (GPU) are designed to perform common operations over large sets of data. GPUs were originally used for driving computer graphics, but this hardware is now used extensively for non-graphics processing. Current applications include machine learning, neural networks, and partial differential equations. To do this, GPUs follow a different paradigm, single instruction-multiple data approach.

With the increasing trend of GPU acceleration for high-performance computing (HPC) clusters, which are traditionally CPU-based, it is crucial to explore the potential benefits and feasibility of leveraging GPU hardware for PFLOTRAN. To assess if PFLOTRAN can utilize GPU hardware in the future, two questions are addressed here. First, do the underlying problems lend themselves to GPU computing? And, if so, how can the existing software be adapted for this use?

## 5.2 Problem Statement

PFLOTRAN is composed of multiple "process models," each of which are first-order differential equations with time. The two eponymous process models are "flow" and "transport". In both cases, there are only two types of fundamental conservation equations to be solved, mass and energy conservation. Expressed in integro-differential form, these are

Mass

$$\frac{d}{dt}\int \phi\rho_i dV + \oint \bar{j}_i \cdot d\bar{S} = 0 \qquad \text{Eq. 5-1}$$

Energy

$$\frac{d}{dt}\int \sum_i \phi\rho_i u_i \, dV + \oint \left(\bar{q} + \sum_i \bar{j}_i\left(u_i + \frac{P}{\rho_i}\right)\right) \cdot d\bar{S} = 0 \qquad \text{Eq. 5-2}$$

Where $\phi$ is porosity, $\rho_i$ is the mass density of component $i$, $\bar{j}_i$ is the mass flux, $u_i$ is the specific internal energy, $P$ is the thermodynamic pressure, and $\bar{q}$ is conductive heat flux. Sources or sinks of mass and energy, if present, can be expressed as additional volume or surface terms. The primary difference between "flow" and "transport" process models is in the definition of the mass flux.

For both flow and transport, PFLOTRAN uses the finite volume method to evaluate the volume and surface integrals. Volume integrals are approximated by using the cell-centered mean value of the integrand. Surface integrals are approximated using mean values at the faces formed between adjacent volumes. In doing so, the finite volume method converts a few integro-differential equations into a large system of ordinary differential equations, one for each conservation equation in each cell.

As the mass and energy balances are only first order with time, they can be solved with either explicit/forward Euler or implicit/backward Euler methods. Implicit is best for deep geologic disposal

modeling because of the large time spans to be simulated. For the implicit method, the set of algebraic, albeit frequently nonlinear, equations to be solved is

$$(\phi\rho_i)^n \frac{V}{\Delta t} + \sum_x \bar{j}_i^n \cdot \bar{S}_x = (\phi\rho_i)^{n-1} \frac{V}{\Delta t} \quad \text{Eq. 5-3}$$

where the objective is to find the next value of porosity and density. The challenge is that the mass flux is itself a function of porosity, density, and energy in the adjacent cells.

The solution is challenging where the number of cells and hence coupled equations is large. Fortunately, while there may be a very large number of ordinary differential equations, relatively few control volumes are adjacent, and the matrix is consequently sparse.

Efficient sparse matrix linear solvers are the provenance of the computational package PETSc used by PFLOTRAN. Development of GPU accelerated linear solver is consequently outside of the scope of PFLOTRAN development and this roadmap.

What is within scope is the efficient evaluation of the mass and energy flux terms that make the mass and energy balances nonlinear. Because they are non-linear, PFLOTRAN relies on iterative techniques, such as Newton's method, to minimize the residual error between the mass and energy across time steps. In Newton's method, it is necessary to not only model the residual error, but also the derivative of the residual error with respect to the independent variables, which is expressed as the Jacobian. Recent code profiling (Park, 2021) indicated that while 31% of the computation time was spent solving the linearized system, 18% was spend on calculating the residuals and 51% was spent on the calculating the Jacobian. The focus here is on reducing the 69% of computational time spent on the residual and Jacobian calculations.

## 5.3 GENERAL flow Model

The GENERAL mode is a two-phase, nonisothermal, two-component, miscible, fully implicit Darcian flow model of PFLOTRAN for water, air, and energy transport. As such, there are three degrees of freedom and three ordinary differential equations for each control volume.

The mass flux of the mixture is typically modeled using the Advective-Dispersive model. The advective flux sum of the products of phase density and velocity.

$$\bar{j}_i = \rho_{i,\alpha}\bar{u}_\alpha + \rho_{i,\beta}\bar{u}_\beta \quad \text{Eq. 5-4}$$

The Darcian or superficial velocity $\bar{u}_\alpha$ is defined as the mean velocity of each phase as if it occupied the entire surface. This Darcian velocity is represented as a function of the intrinsic permeability, the relative permeability and viscosity of the phase, and the phase specific pressure.

$$\bar{u}_\alpha = -k_\infty \frac{k_{r,\alpha}}{\mu_\alpha} \nabla P_\alpha \quad \text{Eq. 5-5}$$

In the limiting case of insoluble phases, the mass flux of the mixture is only dependent upon the velocity of a single phase. To model air solubility in liquid water and water vapor content in the gas phase, additional thermodynamic models are necessary to find the phase density of each species in both phases. To find the phase density, PFLOTRAN applies the local equilibrium model. The local equilibrium model requires that the pressure, temperature, and chemical potential of the two phases be equal within a control volume. Capillary pressure can be used to express the difference (i.e., capillary pressure is idiosyncratic).

$$P_g = P_\ell + P_c \quad \text{Eq. 5-6}$$

$$T_g = T_\ell \quad \text{Eq. 5-7}$$

$$\mu_{w,g} = \mu_{w,\ell} \quad \text{Eq. 5-8}$$

$$\mu_{a,g} = \mu_{a,\ell} \quad \text{Eq. 5-9}$$

For air and water systems, only two thermodynamic equilibrium relationships are necessary. The first is from vapor-liquid equilibrium which states that, for a pure liquid, the fugacity is equal to the vapor pressure of the liquid. For an ideal gas, the fugacity is equal to the partial pressure. For an ideal liquid solution, the vapor pressure is proportional to its mole fraction. Finally, for non-ideal liquid solutions, the vapor pressure may require additional activity coefficient corrections, such as for brine solutions. For this discussion, it is assumed that the air dissolved in the liquid phase negligibly influences the vapor pressure of water.

$$p_{w,g} = x_{w,\ell} p_w^{sat}(T) \cong p_w^{sat}(T) \quad \text{Eq. 5-10}$$

For air and other permanent gases, Raoult's Law cannot be applied as the vapor pressure is undefined above the critical temperature. That is, if any permanent gas is in the liquid phase, it is necessarily due to non-ideal interactions. Fortunately, Henry's Law is a simple and appropriate model for dilute sorbed gases in liquid water solutions:

$$\rho_{a,\ell} = K_{a,\ell} p_{a,g} m_a \quad \text{Eq. 5-11}$$

A common approach in hydrologic models is to express the mixture density using the volumetric degree of liquid saturation, $\theta_\ell$. This is defined as being the volume of the liquid phase under a given condition divided by the total pore volume. For a two-phase gas plus liquid system, this is the sum of the liquid and gas volumes.

$$\theta_\ell = \frac{V_\ell}{V_\ell + V_g} \quad \text{Eq. 5-12}$$

If (and here's the trick) the density of the component in the liquid and gas phase are known, the mixture density of a given component then follows:

$$\rho_i = \rho_{i,\ell}\theta_\ell + \rho_{i,g}(1 - \theta_\ell) \quad \text{Eq. 5-13}$$

## 5.4 Primary Variable Switching

For simulations that are consistently in a two-phase state, the volumetric degree of liquid saturation has been used as one of the "primary" variables in lieu of mixture density. While convenient for consistently unsaturated/two-phase flow, this has two weaknesses. The first is that that volumetric saturation is degenerate at zero and one. Dry and humid air both have a volumetric liquid saturation of zero. Similarly, the dissolved air content in a liquid saturated system cannot be expressed by a volumetric liquid saturation of one. If the simulation enters these states, PFLOTRAN and similar models perform what is known as "primary variable switching". This primary variable switching is summarized below as pseudo-code:

IF $\theta_l = 0$ THEN ! Superheated water vapor

! Primary variables pressure $P$, temperature $T$, and water vapor mole fraction $y_{w,g}$:

$$\frac{d}{dt}\int \phi \rho_{w,g} dV + \oint \rho_{w,g} \bar{u}_g \cdot d\bar{S} = 0$$

$$\frac{d}{dt}\int \phi \rho_{a,g} dV + \oint \rho_{a,g} \bar{u}_g \cdot d\bar{S} = 0$$

$$\rho_{w,g} = m_w x_{w,g} \frac{P}{RT}$$

$$\rho_{a,g} = m_a\left(1 - x_{w,g}\right)\frac{P}{RT}$$

IF $\theta_l = 1$ THEN ! Subcooled liquid water

! Primary variables pressure $P$, temperature $T$, and dissolved air mass fraction $w_{a,\ell}$:

$$\frac{d}{dt}\int \phi\rho_{w,\ell}dV + \oint \rho_{w,\ell}\overline{u}_\ell \cdot d\overline{S} = 0$$

$$\frac{d}{dt}\int \phi\rho_{a,\ell}dV + \oint \rho_{a,\ell}\overline{u}_\ell \cdot d\overline{S} = 0$$

$$\rho_{w,\ell} = (1 - x_{a,\ell})\,\rho_w(P,T)$$
$$\rho_{a,\ell} = x_{a,\ell}\rho_{w,\ell}$$

ELSE ! Vapor-liquid equilibrium

! Primary variables pressure $P$, temperature $T$, and degree of liquid saturation $\theta_\ell$:

$$\frac{d}{dt}\int \phi\left(\rho_{w,\ell}\theta_\ell + \rho_{w,g}(1 - \theta_\ell)\right)dV + \oint \left(\rho_{w,\ell}\overline{u}_g + \rho_{w,g}\overline{u}_g\right) \cdot d\overline{S} = 0$$

$$\frac{d}{dt}\int \phi\left(\rho_{a,\ell}\theta_\ell + \rho_{a,g}(1 - \theta_\ell)\right)dV + \oint \left(\rho_{a,\ell}\overline{u}_g + \rho_{a,g}\overline{u}_g\right) \cdot d\overline{S} = 0$$

$$\rho_{w,\ell} = \rho_w^{sat}(T)$$
$$\rho_{w,g} = \rho_g^{sat}(T)$$
$$\rho_{a,g} = m_a\frac{P - p_w^{sat}(T)}{RT}$$
$$\rho_{a,\ell} = K_{a,\ell}\rho_{a,g}$$

END IF

An advantage of this approach is that the single-phase models are simply compared with the two-phase model, reducing the number of floating-point calculations. The disadvantage is, because it is not known *a priori* when the simulation will transition from a two-phase to one-phase state, any iterations that predict this transition must be discarded and either the time-step size reduced, or the problem reformulated in alternative "primary" variable terms. Consequently, the logic to correctly implement the switching is complex and requires forward and backwards thermodynamic relationships that are precise inverses.

As relates to implementing GPU acceleration, both the variables and equations applied in any given cell now vary conditionally. Therefore, PFLOTRAN presently checks each cell on each iteration, individually, to select the necessary instructions. This approach does not lend itself to efficient implementation on GPUs.

The second problem is not related to GPU acceleration but is related to dry-out conditions in a deep geologic disposal system. The previous definition of liquid saturation assumes that the capillary pressure or matrix potential negligibly perturbs the density of the phases. This approximation is largely acceptable for the liquid phase (though not entirely true, as adsorbed water is 10-20% more dense than liquid water). This approximation is, however, unacceptable for the gas phase. Through the Kelvin equation, it is known that the vapor pressure of a wetting fluid is reduced by the capillary pressure or matric potential. However, the description is incomplete, as it is not only the vapor pressure that is reduced – via a change in the fugacity coefficient – but it is also reduced by the vapor density. The problem here is that the mixture

density is a non-linear function of volumetric liquid saturation. That is, applying the Kelvin equation, the mixture density of water is more precisely stated as

$$\rho_w = \rho_{w,\ell}\theta_\ell + \rho_{w,v}^{sat} e^{-\frac{\psi(\theta_\ell) m_w}{\rho_{w,\ell} RT}} (1-\theta_\ell) \quad \text{Eq. 5-14}$$

If the maximum capillary pressure for silicate materials is ~1 GPa, and the density of liquid water is ~1000 kg/m$^3$, the density of water vapor is reduced at 298 K by a factor of 1000. Consequently, the mixture density of water is nearly zero when the volumetric degree of liquid saturation is zero. Unlike a steam pipe, there is virtually no region where the simulation would be in single-phase gas flow.

Even with the Kelvin equation, water vapor pressure is not accurately represented in porous media below the liquid dry-out limit because of the existence of a third phase, the adsorbed phase. Unlike bulk liquid precipitation, condensation/adsorption onto solids does not have a critical density. Consequently, there is a single gas-phase state with water vapor in porous materials, the system just transitions from a vapor-liquid system to a gas-adsorbate system. Hence, the mixture density is the sum of liquid, vapor, and adsorbed phase:

$$\rho_i = \rho_{i,\ell}\theta_\ell + \rho_{i,g}(1-\theta_\ell) + \rho_{i,s}\frac{(1-\phi)}{\phi} \quad \text{Eq. 5-15}$$

The quantity of adsorbate is proportional to the volume fraction of the solid, such that in the limit of a void region, there is no adsorbate phase.

Now, under liquid phase dry-out conditions defined as the point where the liquid volume is zero, the mixture density can be modeled using an adsorption isotherm. For example, in the Henry's law region:

$$\rho_i = \rho_{i,g} + \frac{1-\phi}{\phi}\rho_s K_{i,s}\rho_{i,g} = (1 + r_{i,s})\rho_{i,g} \quad \text{Eq. 5-16}$$

Where the mixture density is a multiple of the gas phase density, and the gas phase density is not a function of volumetric liquid saturation.

## 5.5 Gravimetric Approach to Eliminate Primary Variable Switching

As an alternative to volumetric saturation, it is proposed that gravimetric saturation be used wherein it is a function of the mixture density relative to the bulk liquid density.

$$\omega = \frac{\rho_w}{\rho_{w,\ell}} \quad \text{Eq. 5-17}$$

The gravimetric saturation is an excellent approximation to the volumetric liquid saturation where the vapor density is negligible compared with the mixture density. In addition, capillary pressure and relative permeability curves are frequently measured gravimetrically, as direct measurement of liquid volume in opaque pores is not feasible. In those cases, using gravimetric saturation removes the approximation implicit in volumetric saturation.

Another advantage is that, for a vectorized model for GPUs, gravimetric saturation is an explicit, linear function of mixture density. This removes the need for primary variable switching and permits an air-water-solid system to be modeled without conditional branching.

Water (Vapor-Liquid-Adsorbate):

$$\frac{d}{dt}\int \phi\rho_w dV + \oint (\rho_{w,\ell}\bar{u}_\ell + \rho_{w,g}\bar{u}_g)\cdot d\bar{S} = 0 \quad \text{Eq. 5-18}$$

$$\rho_{w,l} = \max\left(\rho_w, \rho_l^{sat}(T)\right) \quad \text{Eq. 5-19}$$

$$\rho_{w,g} = \min\left(\frac{\rho_w}{1+r}, \rho_v^{sat}(T) e^{-\frac{\Psi_m(\omega) m_w}{\rho_l^{sat}(T) RT}}\right) \quad \text{Eq. 5-20}$$

Air (Gas-Solute):

$$\frac{d}{dt}\int \phi \rho_a dV + \oint (n_{a,\ell}\overline{u}_\ell + n_{a,g}\overline{u}_g) \cdot d\overline{S} = 0 \quad \text{Eq. 5-21}$$

$$n_{a,g} = \frac{n_a}{1-(1-K_a)\min(\omega,1)} \quad \text{Eq. 5-22}$$

$$n_{a,l} = K_a n_{a,g} \quad \text{Eq. 5-23}$$

Energy (Water + Air):

$$\frac{d}{dt}\int \rho_s U_s(T) + \phi(\rho_a U_a(\rho_a,T) + \rho_w U_w(\rho_w,T))dV + \oint \left(\rho_{a,\ell}\overline{u}_\ell U_\ell + \rho_{a,g}\overline{u}_g U_g + P(\overline{u}_\ell + \overline{u}_g)\right) \cdot d\overline{S} = \dot{Q} \quad \text{Eq. 5-24}$$

$$U_a(\rho_a,T) = \rho_a c_v T - \rho_{a,\ell} U_{absorption}(T) \quad \text{Eq. 5-25}$$

$$U_w(\rho_w,T) = \rho_{w,\ell} U_{w,\ell} + \rho_{w,g} U_{w,g} \quad \text{Eq. 5-26}$$

Here, the phase density applicable to advection (and diffusion) is calculated using intrinsic minimum and maximum functions. These simple conditional instructions can be evaluated on a vector basis as they are members of the intrinsic instruction set and take the exact same number of clock cycles regardless of outcome.

While it seems counterintuitive to permit the density coefficient for liquid advection to be that of saturated liquid, even at gas saturation, the Darcian velocity of the liquid will be zero in such a scenario. Similarly, while the density coefficient for gas advection will be non-zero at liquid saturation, the Darcian gas velocity will also be zero. Flow of the absent phase will cease regardless of the density coefficient. The role of the phase density is to provide the density as if it were present and allow the relative permeability to determine if it is present or absent.

## 5.6 Refactoring PFLOTRAN to Iterate over Material Blocks

While thermodynamic properties of fluids are consistent throughout the domain, the intermolecular potential with the solid varies with each material block. Similar to primary variable switching, the calls to capillary pressure and relative permeability are considered on a cell by cell or scalar basis. While extremely flexible, in most models, the number of material blocks is far smaller than the number of cells. Thus, instead of vectorizing all cells, a good approach would be to iterate over material regions, and vectorize only within a region (i.e., each region has its own $\Psi_m(\omega)$, $k_{r,\ell}(\omega)$, and $k_{r,g}(\omega)$ ). Only one region can be vectorized at a time. At the same time, this can all be combined into one subroutine call as these properties are fundamentally related and reuse many of the expensive exponentiation values as intermediate calculations.

In addition, by refactoring the code from iterating across material blocks instead of cell, the model could also specify different momentum models in different material regions. For example, the Forchheimer equation can be utilized to model flow in fracture material blocks, while Knudsen/Klinkenberg slip flow can be applied in a microporous matrix, without calculating both in all cells. The only requirement in the flow model is that the flow through interfacing surfaces is in consistent terms.

## 5.7 Pressure as Derived Variable

The total mechanical pressure within a volume is the sum of the component pressures. Typically, the water contribution is equal to the saturation pressure, reduced by the Kelvin equation, but it can be less if there is insufficient material to sustain the pressure. If the system is compressed/subcooled liquid, the mechanical pressure of the liquid is equal to saturation plus that stress due to isothermal compression, and the air pressure based on its apparent density in the gas phase. Note, that the gas phase density used is the density the gas would be if the gas phase existed. The dissolved air continues to exert a pressure even when the system is liquid saturated:

$$P = \min\left(p_w^{sat}(T)e^{-\frac{\Psi_m(\omega)}{n_w^{sat}RT}}, \frac{n_w RT}{1+r}\right) + \max\left(0, \frac{\omega - 1}{\beta}\right) + p_a\left(\frac{n_a}{1-(1-K_a)\min(\omega,1)}, T\right) \quad \text{Eq. 5-27}$$

Viscosity (liquid and gas) (these functions can obviously be vectorized)

$$\mu_\ell(T) \quad \text{Eq. 5-28}$$

$$\mu_g(n_{w,g}, n_{a,g}T) \quad \text{Eq. 5-29}$$

Relative permeability (liquid and gas)

$$k_{r,\ell}(\omega) \quad \text{Eq. 5-30}$$

$$k_{r,g}(\omega) \quad \text{Eq. 5-31}$$

Capillary pressure (Matric potential)

$$\Psi_m(\omega) \quad \text{Eq. 5-32}$$

The capillary pressure and relative permeability should be bundled into one subroutine to both ensure consistency and reduce the number of exponentiation operations.

## 5.8 Vectorizing PFLOTRAN

The present software design does not lend itself to vectorization for GPU acceleration. Several design changes are necessary, both to PFLOTRAN to speed the residual and Jacobian calculation and changes to the solvers in PETSc. The PETSc project has its own GPU roadmap to use modern hardware (https://petsc.org/release/overview/gpu_roadmap/)

The PFLOTRAN roadmap to implement GPU acceleration can be broken down into 6 tasks:

1. Eliminate primary variable switching by using a gravimetric approach.
2. Vectorize bulk fluid functions (i.e., water and air equations of states and viscosity).
3. Vectorize and bundle capillary pressure and relative permeability functions.
4. Refactor residuals to be calculated on a material block basis (i.e., call multiple cells at the same time).
5. Test vectorized PFLOTRAN code.
6. Implement GPU versions of the vectorized PFLOTRAN code (with, for example, OpenCL or CUDA).

While the emphasis here is on a density-based flow mode to vectorize the computations for GPU acceleration, there are other potential advantages. Eliminating primary variable switch eliminates discarded

iterations and time-step cutting for systems near either saturation limit. Also, eliminating primary variable switching removes the risk that the forward and backward functions are not exact inverses. Even the relatively limited vector computations supported on CPU hardware may result in modest speed up. (i.e., if eight cells can be evaluated in parallel with AVX-512, this would drop the residual and Jacobian time to a small fraction of the linear solution time).

A final challenge in applying GPU programming is that presently limited support exists for GPU acceleration with ISO Fortran. Currently, two approaches exist to program for GPU's. The first is with OpenCL, which is a C language wrapper for GPU function calls. OpenCL is a framework for C99, C++14 and C++17 programming languages to use not only GPU hardware, but also Digital Signal Processors (DSP) and Field Programmable Gate Array (FPGA) hardware. The OpenCL standard is not tied to any given hardware. Thus, the advantage with OpenCL is that it is non-proprietary and more adaptable to multiple vendor's hardware. However, there is added complexity in implementing non-Fortran modules into the Fortran PFLOTRAN code. Use of OpenCL, being derived from C, requires a different skill set than most PFLOTRAN developers have.

The second approach is to use the NVIDIA-proprietary CUDA Fortran wrapper. This approach would deviate less from Fortran in PFLOTRAN but using CUDA limits the GPU modules to NVIDIA hardware and additionally requires testing the NVIDIA Fortran compiler, nvfortran.

Consequently, the best approach would be to develop a vectorized PFLOTRAN by leveraging what CPU vector operations we can use in standard Fortran and then, if warranted, developing a PFLOTRAN version that uses either CUDE or OpenCL.

# 6. OUTREACH

## 6.1 Introductory Short Course at University of Texas at Austin

In FY24, PFLOTRAN developers from Sandia National Laboratories and Pacific Northwest National Laboratory, in collaboration with the University of Texas at Austin's Bureau of Economic Geology and the Nuclear Engineering program in the Mechanical Engineering Department, conducted a 4-day in-person PFLOTRAN short course from May 7th to May 10th. This introductory course was redesigned since the last in-person session was held in 2018. The course was led by Heeho Park, Glenn Hammond, David Fukuyama, and Rosie Leone; 31 students from across the United States participated.

Participants included researchers from Sandia National Laboratories, Oak Ridge National Laboratory, the University of Texas (covering petroleum, nuclear, and mechanical engineering departments), the Bureau of Economic Geology, and the Jackson School of Geosciences (Figure 6-1). The course covered topics such as fluid flow and chemical transport modeling, multiphase flow and heat transport, radionuclide transport, and visualization. These concepts were integrated into a final practice problem focused on modeling a subsurface radioactive waste repository system. The detailed agenda is provided in Appendix B.


**Figure 6-1. PFLOTRAN short course held at University of Texas at Austin May 2024.**

## 6.2 Global User Base

In FY24, we gathered user engagement data from the PFLOTRAN documentation website Google Analytics to demonstrate that the PFLOTRAN user base is multinational. The top countries with the most users are as follows: United States, China, United Kingdom, South Korea (new in the top seven), Germany, Canada, and Japan. The virtual short course, presentations at conferences, and word of mouth have significantly contributed to building a broad international community. Active users have increased by 93% this year compared to last year, and new user visits have also increased by 96% over the same period. The average engagement time on the documentation page is 9 minutes and 45 seconds, indicating meaningful visits. To visually represent this global engagement, Figure 6-2 presents a world map illustrating user distribution. Countries are shaded from grey, indicating no visits, to dark blue, signifying a high number of visitors. This map vividly showcases the widespread adoption and international reach of PFLOTRAN, reflecting our commitment to fostering a vibrant and connected global scientific community of porous media flow and reactive transport.

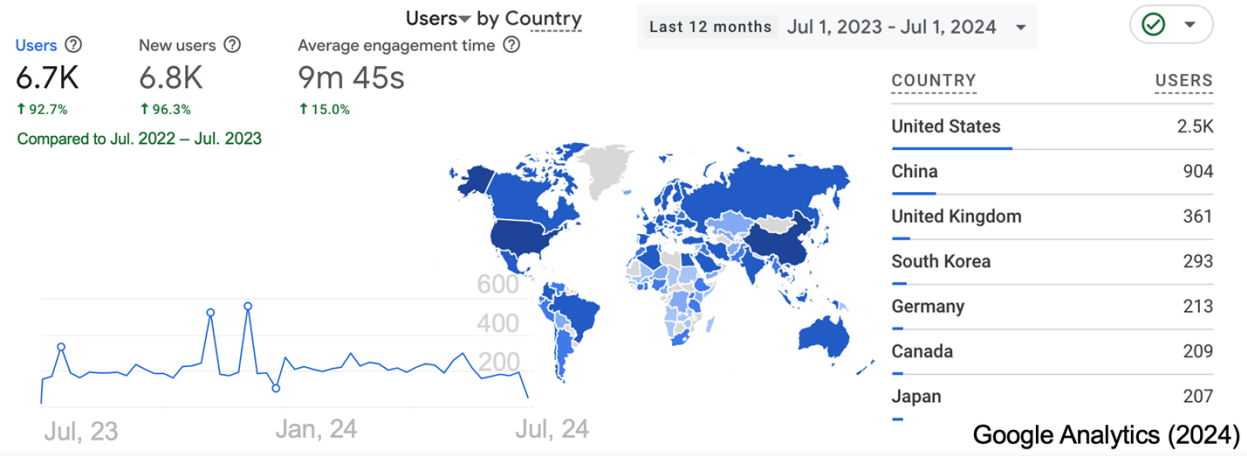



**Figure 6-2. PFLOTRAN user engagement data.**

# 7. REFERENCES

Alt-Epping, P., L. W. Diamond, C. Wanner, G. E. Hammond (2021). Effect of glacial/interglacial recharge conditions on flow of meteoric water through deep orogenic faults: insights into the geothermal system at Grimsel Pass, Switzerland. *Journal of Geophysical Research: Solid Earth*, *126*(7), e2020JB021271.

Balay, S., S. Abhyankar, M.F. Adams, S. Benson, J. Brown, P. Brune, K. Buschelman, E. Constantinescu, L. Dalcin, A. Dener, V. Eijkhout, 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. 2022. *PETSc/TAO User's Manual.* Argonne National Laboratory, ANL-21/39 – Revision 3.17.

Balay, S., J. Brown, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. Curfman McInnes, B. F. Smith, H. Zhang (2013). PETSc User`s Manual. Argonne, Illinois, Argonne National Laboratory.

Chang, K. W., M. Nole, E. R. Stein (2021). Reduced-order modeling of near-field THMC coupled processes for nuclear waste repositories in shale. *Computers and Geotechnics*, *138*, 104326.

Chen, X., G. Hammond, C. Murray, M. Rockhold, V. Vermeul, J. Zachara (2013). "Applications of ensemble-based data assimilation techniques for aquifer characterization using tracer data at Hanford 300 area." *Water Resources Research* 49: 7064-7076.

DOE (2012). Used Fuel Disposition Campaign Disposal Research and Development Roadmap. U.S. DOE Office of Nuclear Energy, Used Fuel Disposition, Washington, D.C.

Frederick, J. M., Hammond, G. E., Stein, E. (2020). *Development of PFLOTRAN Transport Capability for Use in the Waste Isolation Pilot Plant Performance Assessment-20545*, SAND2020-1504C. Sandia National Laboratories, Albuquerque, NM.

Frederick, J. M., (2021). PFLOTRAN v3.0.1 Design Document and User's Manual, ERMS 583102. Sandia National Laboratories, Carlsbad, NM.

Hammond, G. E., J. M. Frederick (2016). *PFLOTRAN Verification: Development of a Testing Suite to Ensure Software Quality*, SAND2016-12589C. Sandia National Laboratories, Albuquerque, NM.

Hammond, G. E., P. C. Lichtner, C. Lu, R. T. Mills (2011). PFLOTRAN: Reactive Flow and Transport Code for Use on Laptops to Leadership-Class Supercomputers. Groundwater Reactive Transport Models. F. Zhang, G. T. Yeh, J. Parker, Bentham Science Publishers.

LaForce T., R. S. Jayne, R. Leone, et al. (2023). DECOVALEX-2023 Task F Specification Revision 10. SAND2023-04005 R. Sandia National Laboratories, Albuquerque, NM.

Frederick, J. M., G. E. Hammond, P. Mariner, E. R. Stein, S. D. Sevougian (2016). *Development of a Waste Form Process Model in PFLOTRAN.* SAND2016-12456C. Sandia National Laboratories, Albuquerque, NM; Sandia National Laboratories, Las Vegas, NV.

Lichtner, P.C., G. E. Hammond, C. Lu, S. Karra, G. Bisht, B. Andre, R. T. Mills, J. Kumar (2013) PFLOTRAN user guide: A massively parallel reactive flow and transport model for describing surface and subsurface processes. Technical report, http://www.bitbucket.org/pflotran/pflotran-dev.

Lichtner, P. C., G. E. Hammond (2012). Quick Reference Guide: PFLOTRAN 2.0 Multiphase-Multicomponent-Multiscale Massively Parallel Reactive Transport Code, LA-CC-09-047. Los Alamos National Laboratory, Los Alamos, NM.

Lu, C., P. C. Lichtner (2007). High resolution numerical investigation on the effect of convective instability on long term CO2 storage in saline aquifers. Journal of Physics Conference Series 78: U320-U325.

Mariner, P., T. Berg, B. Debusschere, A. Eckert, J. Harvey, T. LaForce, et al. (2021). *GDSA Framework Development and Process Model Integration FY2021*. SAND2021-12626R. Sandia National Laboratories, Albuquerque, NM.

Mariner, P.E., Curry, C.J., Debusschere, B.J., Fukuyama, D.E., Harvey, J.A., Leone, R.C., Mendez, C.M., Prouty, J.L., Rogers, R.D., Swiler, L.P. (2023), *GDSA Framework Development and Process Model Integration FY2023*, SAND2023-10906R

Mills, M. M., K. L. Kuhlman, R. S. Jayne, J. B. Coulibaly, B. Reedlunn (2024). *Salt International Collaborations FY24 Update*. SAND2024-XXXX. Sandia National Laboratories, Albuquerque, NM.

Mills, R., C. Lu, P. C. Lichtner, G. E. Hammond (2007). Simulating subsurface flow and transport on ultrascale computers using PFLOTRAN. 3rd Annual Scientific Discovery through Advanced Computing Conference (SciDAC 2007), Boston, Journal of Physics Conference Series.

Navarre-Sitchler, A., R. M. Maxwell, E. R. Siirila, G. E. Hammond, P. C. Lichtner (2013). Elucidating geochemical response of shallow heterogeneous aquifers to $CO_2$ leakage using high- performance computing: implications for monitoring $CO_2$ sequestration. Advances in Water Resources 53: 44-55.

Neretnieks, I., L. Moreno, L. Liu (2017). Clay erosion – impact of flocculation and gravitation. Technical report TR-16-11. Svensk Kärnbränslehantering AB, Swedish Nuclear Fuel and Waste Management Co, Stockholm, Sweden.

Nole, M., G. D. Beskardes, D. Fukyuama, R. C. Leone, H. D. Park, M. Paul, A. Salazar, G. E. Hammond, P. C. Lichtner (2023). *Recent Advancements in PFLOTRAN Development for the GDSA Framework*. SAND2023-07655R. Sandia National Laboratories, Albuquerque, NM.

Nole, M., G. D. Beskardes, D. Fukuyama, R. C. Leone, P. Mariner, H. D. Park, M. Paul, A. Salazar, G. E. Hammond, P. C. Lichtner (2022). *PFLOTRAN Development FY2022*, SAND2022-10526R. Sandia National Laboratories, Albuquerque, NM.

Nole, M., R. C. Leone, H. D. Park, M. Paul, A. Salazar, G. E. Hammond, P. C. Lichtner (2021). GDSA PFLOTRAN Development (FY2021), SAND2021-8709R. Sandia National Laboratories, Albuquerque, NM.

Park, H. D., G. E. Hammond, A. J. Valocchi, T. LaForce (2021). Linear and nonlinear solvers for simulating multiphase flow within large-scale engineered subsurface systems. *Advances in Water Resources*, *156*, 104029.

Peaceman, Donald W. (1978) Interpretation of well-block pressures in numerical reservoir simulation (includes associated paper 6988). Society of Petroleum Engineers Journal 18, no. 03: 183-194.

Posiva (2013). Safety Case for the Disposal of Spent Nuclear Fuel at Olkiluoto - Models and Data for the Repository System 2012. POSIVA 2013-01. Posiva Oy, Olkiluoto, Eurajoki, Finland.

Price, L. L., A. Alsaed, A. C. Barela, E. Basurto, J. C. Prouty, A. Salazar, C. Sanders, A. Taconi, M. Swinney, F. Davidson, N. Kucinski, N. Panicker, E. Gonzalez, B/ Kiedrowski (2022) *Effects of Postclosure Criticality on Repository Performance*, SAND202210297 R, Sandia National Laboratories, Albuquerque, NM.

Sassani, D., J. Birkholzer, C. Camphouse, G. Freeze, E. R. Stein (2021). *SFWST Disposal Research R&D 5-Year Plan-FY2021 Update*. SAND2021-12491R. Sandia National Laboratories, Albuquerque, NM.

Sevougian, S. D., P. E. Mariner, L. A. Connolly, R. J. MacKinnon, R. D. Rogers, D. C. Dobson, J. L. Prouty (2019). DOE SFWST Campaign R&D Roadmap Update, Rev. 1. Sandia National Laboratories, Albuquerque, NM.

Stein, E. R., J. M. Frederick, G. E. Hammond, K. L. Kuhlmann, P. E. Mariner, S. D. Sevougian 2017, April 9-13, 2017. *Modeling Coupled Reactive Flow Processes in Fractured Crystalline Rock*. Paper presented at the International High-Level Radioactive Waste Management Conference, Charlotte, NC.

Swiler, L.P., Brooks, D.M., Portone, T., Basurto, E., Mariner, P.E., Leone, R.C. (2023), *Uncertainty and Sensitivity Analysis Methods and Applications in the GDSA Framework (FY2023)*, Sandia National Laboratories, Albuquerque, NM, SAND2023-08550R.

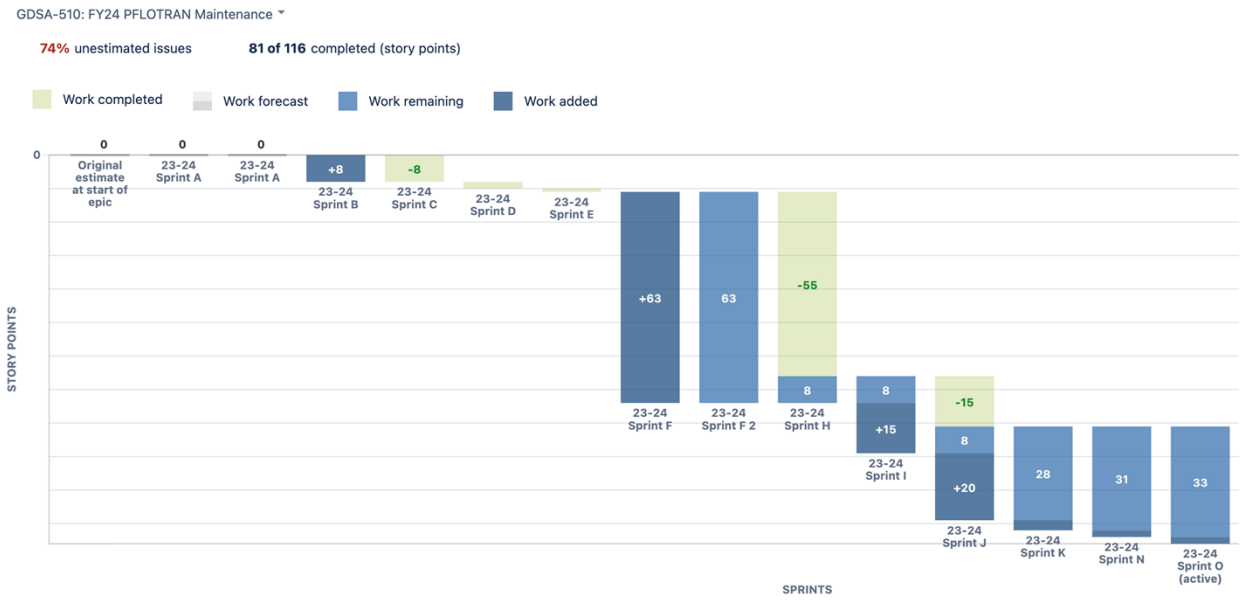
White, M. D., T. J. Kneafsey, Y. Seol, W. F. Waite, S. Uchida, J. S. Lin, et. al. (2020). An international code comparison study on coupled thermal, hydrologic and geomechanical processes of natural gas hydrate-bearing sediments. *Marine and Petroleum Geology*, *120*, 104566.

Wu, R., X. Chen, G. E. Hammond, G. Bisht, X. Song, M. Huang, et al. (2021). Coupling surface flow with high-performance subsurface reactive flow and transport code PFLOTRAN. *Environmental Modelling & Software*, *137*, 104959.

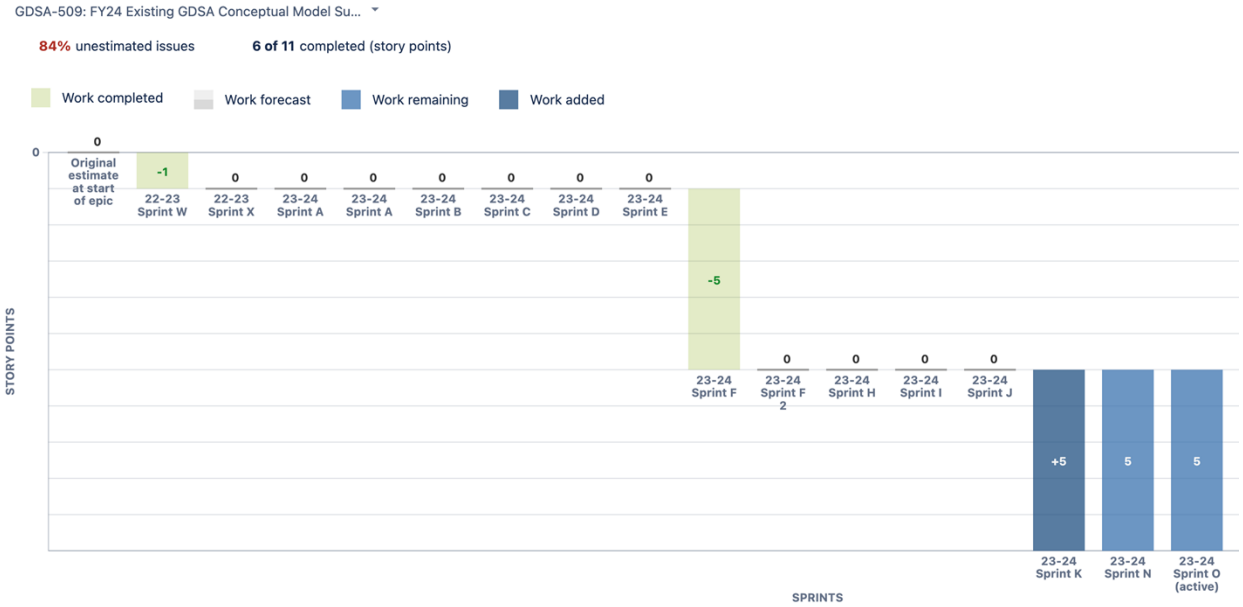
# APPENDIX A: SOFTWARE DEVELOPMENT ISSUES RESOLVED

Appendix A categorizes 141 issues created and 129 issues resolved between August 2023 to July 2024 (past 365 days) into four general categories: (1) PFLOTRAN maintenance, (2) support for existing GDSA conceptual models, (3) refinements to the buffer erosion/canister corrosion model, and (4) enhancements to salt mode.

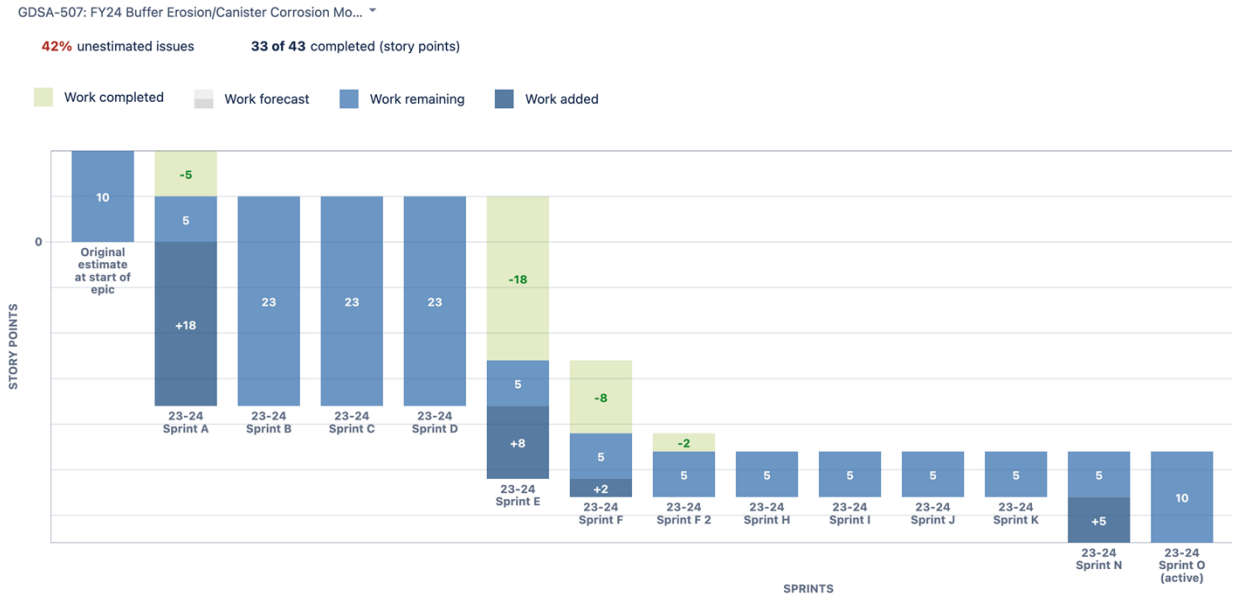
## A.1 FY24 PFLOTRAN Maintenance



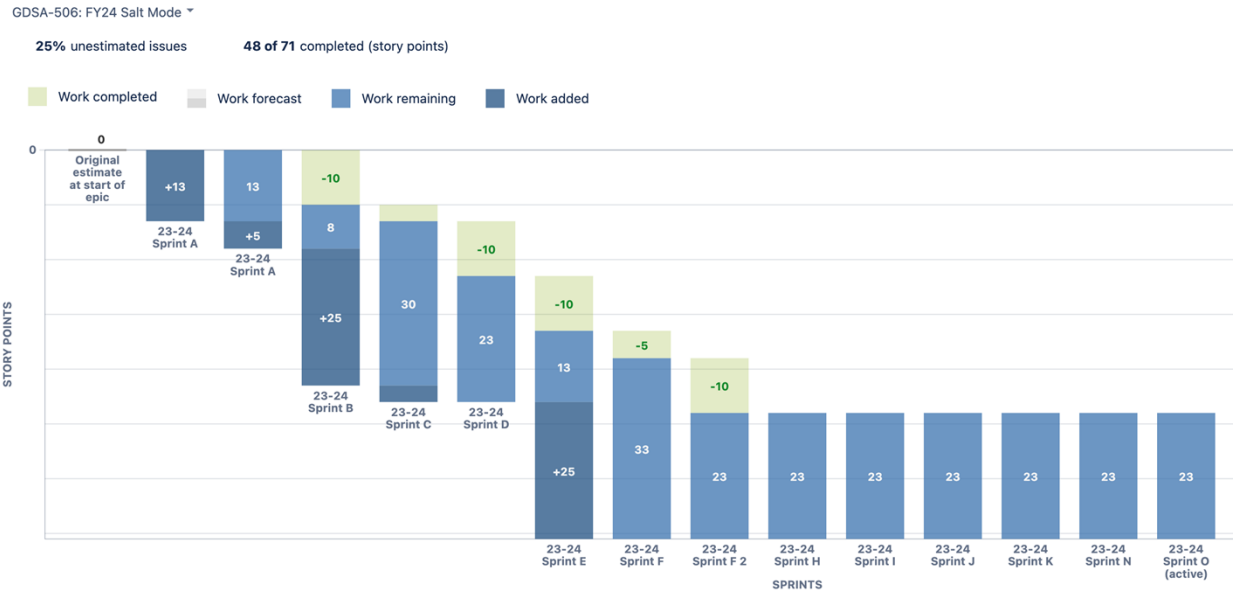

## A.2 FY24 GDSA Conceptual Model Support



## A.3 FY24 Buffer Erosion/Canister Corrosion Process Model




## A.4 FY24 Implicit Salt Transport Option



# APPENDIX B. BUFFER EROSION COPPER CORROSION ADDITIONAL MATERIALS

| Parameter | Description | Value |
|---|---|---|
| FRACTURE_ANGLE | Angle of largest fracture intersecting borehole, specified in WASTE_FORM block | 1.5708 rad |
| FRACTURE_APERTURE | Aperture of largest fracture intersecting borehole, specified in WASTE_FORM block | 0.0001 m |
| WATER_VELOCITY | Water velocity across borehole, specified in WASTE_FORM block | 41 m/yr |
| ION_CONCENTRATION | Salinity of groundwater at borehole location, specified in WASTE_FORM block | 1.946 M |
| SMECTITE_PARTICLE_DENSITY | Buffer grain density | 2700 kg/m$^3$ |
| BOREHOLE_RADIUS | Radius of borehole | 0.925 m |
| DIFF_COEF_IN_BH | Diffusion coefficient in borehole | $1\times10^{-9}$ m$^2$/s |
| SMECTITE_VOL_FRAC_BH_INT | Smectite volume fraction in borehole initially | 0.574 |
| SMECTITE_VOL_FRAC_AT_RIM | Smectite volume fraction at intruding rim | 0.015 |
| Y0 | Time-dependent empirical exponent $y_0$ for total buffer extruded over time (Eq. 4.3, Neretnieks et al. 2017) | 93.74 |
| Y1 | Time-dependent empirical exponent $y_1$ for total buffer extruded over time (Eq. 4.3, Neretnieks et al. 2017) | -0.0004521 |
| Y2 | Time-dependent empirical exponent $y_2$ for total buffer extruded over time (Eq. 4.3, Neretnieks et al. 2017) | $2.236\times10^{-9}$ |
| CI_UPPER_BOUND | Salinity threshold above which no buffer is eroded. | 4 mM |
| CI_LOWER_BOUND | Salinity lower bound for smectite diffusivity calculation. | 0.1 mM |
| SEDIMENT_RELEASE_CONSTANT | Experimentally determined sedimentation rate | 1000 kg/(m$^2$-yr) |

| Parameter | Description | Value |
|---|---|---|
| AGGLOMERATE_FLUID_VISCOSITY | Viscosity of agglomerate fluid | 0.1 Pa-s |
| AF_DENSITY | Density of agglomerate fluid | 1017 kg/m$^3$ |
| AF_VOL_DEN | Volume density of agglomerate fluid | 0.01 |
| F1 | Volume multiplier to critical volume of buffer loss to expose copper | 1 |
| BUFFER_POROSITY | Porosity of buffer | 0.35 |
| CANISTER_RADIUS | Radius of canister | 0.525 m |
| CANISTER_DENSITY | Density of canister material | 8900 kg/m$^3$ |
| CANISTER_WALL_THICKNESS | Thickness of canister wall | 0.047 m |
| REACTANT_CONC_HS | Reactant concentration | $1.21\times10^{-4}$ M |
| MW_CANISTER_METAL | Molecular weight of canister material | 63.54 g/mol |
| MW_REACTANT | Molecular weight of corroding agent | 33.07 g/mol |
| METAL_TO_REACTANT_RATIO | Stoichiometric ratio of corroding agent to corroding metal | 2 |
| EXPOSED_SURFACE_MULTIPLIER | Exposed surface multiplier to area of copper exposed after buffer erosion | 1 |

# APPENDIX C. PFLOTRAN INTRODUCTORY SHORT-COURSE AGENDA

**Short Course University of Texas - Austin**

**Location:** The BEG's VR Room, Bldg 130, Rm 1.116C, JJ Pickle Research Campus, 0611 Exploration Way, Austin, TX 78758

Instructors: Heeho Park, David Fukuyama, Rosie Leone, and Glenn Hammond

## Tuesday, May 7, 2024 (Central Time)

| | |
|---|---|
| 9:00am – 9:15am | *Greetings* |
| 9:15am – 10:00am | [Glenn][p] PFLOTRAN Overview |
| | *Break* |
| 10:10am – 11:15am | [Heeho, Rosie, David][p] PFLOTRAN Use cases - Repository Safety Analyses |
| 11:15am – 11:45am | [Rosie][e] VM or Linux/MacOS setup |
| 11:45am – 1:00pm | *Lunch* |
| 1:00pm – 1:30pm | [Heeho][p] PFLOTRAN Flow |
| 1:30pm – 2:00pm | [David][p] PFLOTRAN input deck - Flow |
| | *Break* |
| 2:15pm – 3:00pm | [Rosie][e] 1D Variably Saturated Flow |
| 3:00pm – 4:00pm | [Heeho][e] Two-phase General Model (3 impermeable block model) |
| | *Break* |
| 4:15pm – 5:00pm | [Glenn][e] Error Messaging |
| 5:00pm – 5:10pm | [Everyone] Q & A |

[p] Presentation
[e] Hands-on Exercise

# Wednesday, May 8 (Central Time)

| | |
|---|---|
| 9:00am – 10:00am | [Rosie][e] Regional Doublet Problem |
| | *Break* |
| 10:15am – 11:00am | [Glenn][p] PFLOTRAN Reactive Transport |
| 11:00am – 12:15pm | [David][p] PFLOTRAN Input deck - Transport<br>[David][e] Calcite reaction |
| 12:15pm – 1:30pm | *Lunch* |
| 1:30pm – 2:30pm | [Glenn][p] Copper Leaching |
| | *Break* |
| 2:45pm – 3:45pm | [David][p] Regional Flow Simulation |
| | *Break* |
| 4:00pm – 5:00pm | [Heeho][e] Paraview |
| 5:00pm – 5:10pm | [Everyone] Q & A |

[p] Presentation
[e] Hands-on Exercise

# Thursday, May 9 (Central Time)

| | |
|---|---|
| 9:00am – 10:15am | [Heeho][p] Unstructured grid and Voronoi Mesh |
| | *Break* |
| 10:30am – 11:45am | [Heeho][e] 3D Flow and Transport |
| 11:45am – 1:00pm | *Lunch* |
| 1:00pm – 2:00pm | [Rosie][e] Geologic Disposal 2D Example problem |
| | *Break* |
| 2:15pm – 3:00pm | [Rosie][e] 1D tracer single fracture network |
| 3:00pm – 5:00pm | [Everyone][e] Kryptonite Challenge problem |

[p] Presentation
[e] Hands-on Exercise

# Friday, May 10 (Central Time)

| | |
|---|---|
| 9:00am – 9:45am | [Heeho][p] Geologic Disposal Safety Assessment Overview |
| 9:45am – 10:30am | [Glenn][p] PFLOTRAN QA |
| | *Break* |
| 10:45am – 12:00pm | [Everyone][e] Continue Kryptonite Challenge problem |
| 12:00pm – 1:00pm | *Lunch* |
| 1:00pm – 4:00pm | NETL TOUR |

[p] Presentation

[e] Hands-on Exercise

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

# APPENDIX D—NFCSC DOCUMENT COVER SHEET

## NFCSC DOCUMENT COVER SHEET[1]

Name/Title of Deliverable/Milestone/Revision No. Recent Advancements in PFLOTRAN Development for the GDSA Framework / M3SF-23SN010304101

Work Package Title and Number GDSA PFLOTRAN Development – SNL / SF-24SN01030410

Work Package WBS Number. 1.08.01.03.04.10

Responsible Work Package Manager Heeho Park (Name/Signature)

Date Submitted 08/15/2024

| Quality Rigor Level for Deliverable/Milestone[2] | ☐ QRL-1 ☐ Nuclear Data | ☐ QRL-2 | ☒ QRL-3 | ☐ QRL-4 Lab QA Program[3] |
|---|---|---|---|---|

This deliverable was prepared in accordance with Sandia National Laboratories
*(Participant/National Laboratory Name)*

QA program which meets the requirements of

☒ DOE Order 414.1 ☐ NQA-1 ☐ Other

**This Deliverable was subjected to:**

☒ Technical Review ☐ Peer Review

**Technical Review (TR)**
**Review Documentation Provided**

☐ Signed TR Report or,

☐ Signed TR Concurrence Sheet or,

☒ Signature of TR Reviewer(s) below

**Peer Review (PR)**
**Review Documentation Provided**

☐ Signed PR Report or,

☐ Signed PR Concurrence Sheet or,

☐ Signature of PR Reviewer(s) below

**Name and Signature of Reviewers**

Kris Kuhlman

Lisa Bigler

**NOTE 1:** *Appendix E should be filled out and submitted with the deliverable. Or, if the PICS:NE system permits, completely enter all applicable information in the PICS:NE Deliverable Form. The requirement is to ensure that all applicable information is entered either in the PICS:NE system or by using the NFCSC Document Cover Sheet. In some cases there may be a milestone where an item is being fabricated, maintenance is being performed on a facility, or a document is being issued through a formal document control process where it specifically calls out a formal review of the document. In these cases, documentation (e.g., inspection report, maintenance request, work planning package documentation or the documented review of the issued document through the document control process) of the completion of the activity, along with the Document Cover Sheet, is sufficient to demonstrate achieving the milestone.*

**NOTE 2**: *If QRL 1, 2, or 3 is not assigned, then the QRL 4 box must be checked, and the work is understood to be performed using laboratory QA requirements. This includes any deliverable developed in conformance with the respective National Laboratory / Participant, DOE or NNSA-approved QA Program.*

**NOTE 3:** *If the lab has an NQA-1 program and the work to be conducted requires an NQA-1 program, then the QRL-1 box must be checked in the work Package and on the Appendix E cover sheet and the work must be performed in accordance with the Lab's NQA-1 program. The QRL-4 box should not be checked.*

U.S. DEPARTMENT OF ENERGY

Sandia National Laboratories