INL/RPT-22-70694
Revision 0

Integration Plan for NEAMS Structural Materials and Chemistry MSR Modeling

December 2022

Benjamin W. Spencer
Idaho National Laboratory

Theodore M. Besmann
University of South Carolina

Idaho National Laboratory

*INL is a U.S. Department of Energy National Laboratory
operated by Battelle Energy Alliance, LLC*

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, trademark, 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.

INL/RPT-22-70694
Revision 0

Integration Plan for NEAMS Structural Materials and Chemistry MSR Modeling

Benjamin W. Spencer
Idaho National Laboratory

Theodore M. Besmann
University of South Carolina

December 2022

Idaho National Laboratory
Originating Organization
Idaho Falls, Idaho 83415

http://www.inl.gov

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

Page intentionally left blank

CONTENTS

FIGURES

ACRONYMS

API	application programming interface
GEM	Gibbs energy minimizer
MOOSE	Multiphysics-Object Oriented Simulation Environment
MOOSE-CR	MOOSE Chemical Reactions (module)
MOOSE-PF	MOOSE PhaseField (module)
MSR	molten salt reactor
MSTDB-TC	Molten Salt Thermal Properties Database-Thermochemical
MSTDB-TP	Molten Salt Thermal Properties Database-Thermophysical
NEAMS	Nuclear Energy Advanced Modeling & Simulation
TH	thermal-hydraulic

Page intentionally left blank

Integration Plan for Structural Materials and Chemistry

1. OVERVIEW

Molten-salt reactors (MSRs) are an important class of advanced nuclear reactors currently being developed for commercial energy generation. MSRs employ salts (fluoride or chloride) as a medium for transferring heat from the reactor core. In many MSR designs, the fuel is dissolved in the salt, although it can also be in solid form in the reactor core. For either type of design, many of the same salt-behavior-related issues must be addressed, including the need to characterize the following:

- Movement of chemical species throughout the salt-based coolant system, including the base salts, fuel materials, fission and transmutation products, impurities introduced by interactions with salt-facing structural materials, and other contaminants. As the local composition strongly affects the reactor physics calculations, species concentrations become important and must thus be defined and considered when assessing reactor system safety and performance.
- Precipitating phases in the system, which are a function of chemical composition and temperature and can affect the salt's thermophysical properties, causing various operational and safety issues.
- Interaction with the solid materials contacting the salt throughout the coolant system. Such materials can include metals in the reactor and piping, graphite channels and reflectors, and graphite fuel pebbles. The phenomena of interest include leaching of chemical species from these solid components, as well as surface deposition or intrusion.
- Degraded performance of structural components in the reactor core and coolant system, due to interactions with molten salts. Material performance is affected by multiple aspects of the reactor environment, including high temperatures, radiation, and corrosion mechanisms.

As part of its overarching objective to support advanced reactor development, the U.S. Department of Energy's Nuclear Energy Advanced Modeling and Simulation (NEAMS) program has a strong interest in developing simulation capabilities to address these issues for MSRs. To that end, the Structural Materials and Chemistry technical area within the NEAMS program is developing a set of codes and databases to describe the behavior of molten salts and their interactions with structural materials. In some cases, this effort utilizes tools developed by other program areas; therefore, all involved codes are discussed in this document.

While these tools will initially be used in research settings as they are being developed, the ultimate objective is for MSR vendors and/or regulators to use them in the regulatory process. Thus, it is important to develop capabilities in a timely manner to ensure their relevance. Capabilities should be incrementally released upon being developed and validated, and viable tools should be made available on a time scale of several years.

A previous document [1] summarized the NEAMS MSR codes, filling the need for further clarification of the specific roles of these various codes, along with the current and planned interfaces between them. This report reviews the current activities with respect to MSR-relevant codes and represents an effort to improve their integration for simulating MSR systems. Interoperability of the codes being developed is crucial to the success of the NEAMS-developed MSR simulation capability. This interoperability is greatly facilitated by basing the codes on the MOOSE framework when possible, and by developing interfaces to MOOSE-based tools for any tools not based on that framework.

2. Current Codes and Databases

Figure 1 shows a graphical depiction of the phenomena that must be accommodated in MSR modeling, along with the interrelationships among them. The NEAMS program is developing capabilities to simulate and characterize these phenomena. The current state of the codes and databases being developed to meet these needs are described in the sections below. It is important to note that, in some cases, the roles of these codes are being adjusted from these descriptions. The integration plan in Section 3 summarizes the changes being made to these codes to better align their development efforts.

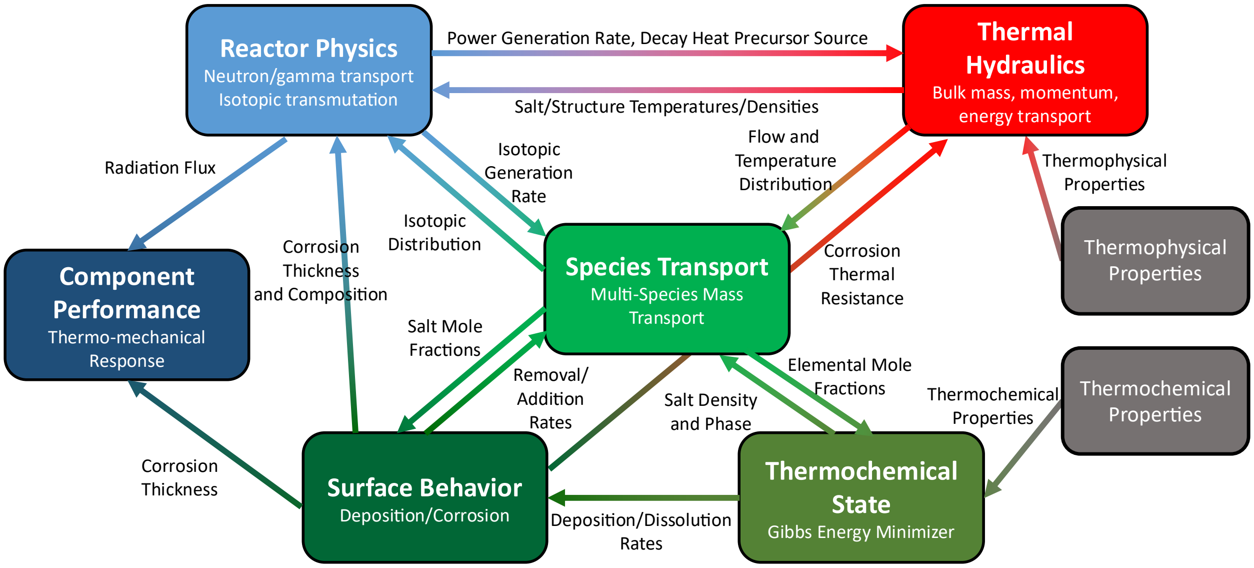

Reactor Physics
Neutron/gamma transport
Isotopic transmutation
Power Generation Rate, Decay Heat Precursor Source
Thermal Hydraulics
Bulk mass, momentum, energy transport
Salt/Structure Temperatures/Densities
Radiation Flux
Isotopic Generation Rate
Flow and Temperature Distribution
Thermophysical Properties
Isotopic Distribution
Corrosion Thickness and Composition
Species Transport
Multi-Species Mass Transport
Corrosion Thermal Resistance
Thermophysical Properties
Component Performance
Thermo-mechanical Response
Salt Mole Fractions
Removal/ Addition Rates
Salt Density and Phase
Elemental Mole Fractions
Thermochemical Properties
Thermochemical Properties
Corrosion Thickness
Surface Behavior
Deposition/Corrosion
Deposition/Dissolution Rates
Thermochemical State
Gibbs Energy Minimizer

Figure 1. Physical phenomena in MSR systems addressed by modeling and simulation codes, along with the interrelationships among them.

2.1 MSTDB-TP

The Molten Salt Thermal Properties Database-Thermophysical (MSTDB-TP) contains sets of referenced values and relations for thermophysical properties, including density, thermal conductivity, viscosity, heat capacity, and related optical properties. The database is maintained as a comma-separated value (CSV) file of salt system thermophysical property data, uncertainty values (where available), and literature references. The database is made available via the Oak Ridge National Laboratory GitLab repository, which is accessible at https://code.ornl.gov upon request to MSTDB@ornl.gov.

Interfaces

- Direct use: The simple format of this database enables it to be used for a variety of purposes by providing direct access to the data.
- Saline: The Saline code provides application programming interfaces (APIs) for multiple programming languages, allowing other codes to programmatically access and use these data. This is expected to be the primary means of accessing the MSTDB-TP data required by the NEAMS codes.

2.2 MSTDB-TC

The Molten Salt Thermal Properties Database-Thermochemical (MSTDB-TC) is a publicly available compendium of internally consistent chemical thermodynamic models and values for fluoride, chloride, and iodide salt components and related systems of interest to MSR technology [2].

- MSTDB-TC provides thermodynamic information in the ChemSage (ASCII) .dat file format for use with the FactSage® commercial thermodynamic codes and the open-source equilibrium code Thermochimica. It will soon also be usable in the pycalphad [3] open-source software, which is being modified to handle the system models. These software packages utilize the Gibbs energy functions provided by MSTDB-TC in global energy minimization calculations to determine the local equilibrium state of a given chemical system.
- Separate files are provided for both chloride- and fluoride-based systems, and each also contain the relevant iodide components.
- The documentation includes:
  - A detailed description of the current database format and current version
  - A spreadsheet tracing all the data sources
  - A library of published phase diagrams—along with companion MSTDB-TC-computed phase diagrams for direct comparison.
- The database is available via the Oak Ridge National Laboratory GitLab repository at https://code.ornl.gov by request to MSTDB@ornl.gov.

Interfaces

- Standalone thermodynamic equilibrium calculations can be performed using FactSage®, Themochimica, or pycalphad.
- The Thermochimica equilibrium code can be coupled to MOOSE codes to perform equilibrium calculations; thus, MSTDB-TC can be directly used in system models via Thermochimica.
- Yellowjacket Gibbs energy minimizer (GEM): The Yellowjacket GEM code is a MOOSE-based, C++ implementation of a GEM and can directly read the ChemSage-formatted data in MSTDB-TC to use those data in its calculations.

2.3 Saline

Saline is an API to MSTDB-TP data and thus a defined interface for interacting with the data stored in MSTDB-TP. Saline extracts the needed data from MSTDB-TP and applies appropriate models to compute the properties for the system of interest. As such, any code can use Saline, though Saline itself does not require or utilize any other codes.

Interfaces

- Direct use: Any code can use Saline's APIs to access MSTDB-TP data. One important use case is Python codes for analyzing and plotting these data.
- SAM: Saline provides molten salt thermal-hydraulic (TH) property data to SAM.
- Yellowjacket (planned): Saline could be used by Yellowjacket phase-field simulations of corrosion in salt-facing components to interface with MSTDB-TP to obtain properties of the salt portion of the domain.
- Mole (planned): Saline could provide thermophysical property data extracted from MSTDB-TP.

2.4 Mole

Mole is a code for modeling multi-species, multi-phase reactive mass transport to account for fission products, transmutation elements, and other salt constituents in the coolant loop, acting as an integrator of local physics models. It is essentially a volumetric balance of all sources and sinks, including corrosion, transmutation, and fission product buildup. Mole acquires local inventory values from other codes (e.g., in the case of corrosion from MOSCATO and/or the MOOSE Chemical Reactions [MOOSE-CR]) module. Mole is built on the MOOSE framework and represents the coolant loop using a series of 1D line elements. This reduced dimensionality allows Mole to efficiently model the entire coolant system, making it a practical tool for engineering computations. Extension to 2D and 3D calculations would be straightforward, and is expected to be needed for some applications.

Interfaces

- SAM: Mole receives TH conditions (velocity, temperature, pressure, density) from SAM. It then sends updated local species concentrations to SAM. "Full coupling" has been demonstrated between Mole and SAM, with the two codes being compiled together and run as a single executable. In the longer term, it will likely be necessary to use the MultiApps approach.
- Pronghorn: Mole receives the 2D/3D near-wall TH fields from Pronghorn, computes the species state, and catalogs composition and state for species tracking.
- Griffin (in progress): Mole could receive burnup and transmutation rates from Griffin, then send updated isotopic distributions back to Griffin. This work is currently ongoing, using delayed neutron precursors as a simple test case for the coupling. A next step is to investigate the possible paths for isotopic transmutation.
- Yellowjacket (planned): Mole could receive rate information from Yellowjacket (e.g., deposition/corrosion rates and phase change rates), then send updated local species concentrations, temperatures, pressures, or other information needed by Yellowjacket.
- Saline: If Mole is being used without SAM coupling, or if it is used coupled with SAM but SAM does not make use of Saline, Mole may use Saline to retrieve thermophysical salt properties.
- GEM: Likely to be used by Mole, either directly or indirectly (via Yellowjacket).

2.5 Pronghorn

Pronghorn is a coarse-mesh TH code. It resolves the 2D/3D TH field in full MSR cores at the engineering scale (centimeters), accounting for weak compressibility in the flow field. Prior reports have covered the efforts made to integrate intermediate-fidelity corrosion models into Pronghorn by utilizing the Poisson-Nernst-Planck model for electrochemical species migration, and to allow for 3D-resolved species tracking in MSRs by using numerical schemes with limiters. However, the corrosion models and subsequent source terms and potential structural material degradation rate are considered better represented by other codes (e.g., MOOSE-CR combined with phase-field calculations, or by the MOSCATO code). In addition, Mole is expected to be utilized for species tracking; therefore, Pronghorn should be informed by Mole and provide local 2D/3D TH information to Mole. When coupled to Griffin, Mole will be able to provide composition information to Pronghorn. Important characteristics of Pronghorn include:

- Numerical methods developed following NQA-1 code development standards
- Coupled neutronics (Griffin) + thermal hydraulics (Pronghorn) verified for high temperature gas reactor, fluoride salt-cooled high-temperature reactor, and molten salt fast reactor benchmarks (there are no experiments for that)
- In active usage and undergoing further development for MSRs, sodium fast reactors, and nuclear thermal propulsion applications.

Interfaces

- SAM: SAM performs 1D-system-level TH models, while Pronghorn is applied in sections needing 2D/3D resolution (e.g., Pronghorn models the primary loop in pool-type MSRs, whereas SAM models the balance of plant).

- Griffin: Griffin performs neutronics calculations using 3D temperature and density fields computed by Pronghorn, and Pronghorn reads nuclear power from Griffin (e.g., model steady-state conditions for a MSR, with temperature feedback to neutronics, or an unprotected transient event).
- Mole: Mole could compute the radionuclide species states and species mass source terms to update the density and species distribution profiles in Pronghorn.
- MOSCATO (planned): Pronghorn could perform coarse-mesh 2D/3D TH analyses for safety studies (computationally inexpensive), whereas MOSCATO could perform wall-resolved 3D TH analyses for detailed TH studies (computationally intensive).
- Saline (planned): Whereas current thermophysical models in Pronghorn rely on correlations, Saline could provide detailed, on-the-fly thermophysical properties to Pronghorn to improve correlation-based results.
- GEM (planned): GEM could provide the detailed thermochemical modeling to provide properties for the Pronghorn solution of 3D temperature and pressure fields.

2.6 MOSCATO

MOSCATO performs multiscale, multiphysics modeling of chemistry and corrosion phenomena in molten salts [4], and is being actively developed to implement new features that enable high-fidelity simulations of a wide range of crucial chemistry and corrosion phenomena. Interfacing to the MSTDBs will be implemented in FY23. No direct coupling to other codes is planned at present, except via the high-fidelity mass transfer correlations MOSCATO directly provides to reduced-order solvers. MOSCATO makes use of alternate, internally generated thermodynamic descriptions of molten salt chemistry, and presently has no relationship to the Thermochimica or GEM software.

Interfaces

- SAM: MOSCATO provides correlations that enable lower-order solvers such as SAM to produce meaningful chemistry/corrosion results.
- Mole: MOSCATO provides correlations in a manner similar to SAM.
- Yellowjacket (planned): MOSCATO could make use of the alloy properties provided via Yellowjacket's mesoscale simulations.
- Saline (planned): The Saline API could be employed as an interface between MOSCATO and MSTDB-TP.

2.7 SAM

SAM is a code for whole-reactor system transient simulation, safety analysis, system-level thermal fluid design and analysis, and system-level mass transport. It is currently in beta release while undergoing continuing development and integration.

Interfaces

- Griffin: Griffin provides a detailed power distribution used in SAM's thermal models.
- MOSCATO: SAM uses reduced-order models (parameters, coefficients) from MOSCATO in its simulations of system-level mass transport.
- Mole (planned): Mole could provide mass source or sink terms due to various physical processes, for use in the system-level mass transport models in SAM.
- Yellowjacket (planned): Yellowjacket could provide parameters for mass transport both to and from salt-facing components.

- Saline: Salt thermophysical properties are provided by Saline using MSTDB-TP data.
- GEM (planned): SAM will, as needed, use either Thermochimica or GEM with MSTDB-TC for determining speciation in source term calculations.

2.8 Corrosion Modeling via Yellowjacket and the MOOSE PhaseField Module

Yellowjacket models the corrosion of molten-salt-facing alloys, including the flux of corrosion products into the salt and the degradation of the alloy properties. Yellowjacket is MOOSE-based and focused on phase-field modeling [5]. It is currently capable of simulating interactions between Hastelloy and FLiNaK, Code development and verification for that problem is complete, and validation is now underway. It accounts for chemical driving forces for structural component leaching into the salt, fast diffusion along grain boundaries, and void growth. Coupling with the GEM code has been demonstrated and continues to be further developed. The expectation is to couple the code to Grizzly and Mole as part of a Nuclear Energy University Programs-supported effort.

Interfaces

- MOSCATO (planned): Yellowjacket will be used to create surrogate models that define the diffusion of corrosion products from the alloy into the salt in a way that is usable by MOSCATO.
- GEM: Used to create polynomial approximations of free energies.
- Grizzly: Yellowjacket will be used to create surrogate models for Grizzly—models that define the corrosion-caused degradation in the Young’s modulus and yield stress.
- Mole: Yellowjacket will be used to create surrogate models for Mole—models that define the diffusion of corrosion products from the alloy into the salt.

2.9 Yellowjacket Gibbs Energy Minimizer

Yellowjacket has a GEM for direct coupling with other MOOSE apps. Although the GEM can be used in a number of ways, the primary goal has been to provide free energies for phase-field modeling aimed at molten-salt corrosion applications. The best way to couple to other codes has yet to be determined—although potential avenues exist for code-coupling via SAM—and extensive standalone use is also expected. Though most of the GEM’s core capabilities are in place, it is not yet operational (expected FY-22). During FY-21, the code underwent a major rewrite to ensure better consistency with other MOOSE apps, but extensive testing is still required.

Interfaces

- Yellowjacket: The GEM will be used to create polynomial approximations of free energies so that they can be used in phase-field models of corrosion and species transport in salt-facing alloys.
- MSTDB-TC: The GEM will use the MSTDB-TC data as its inputs.

2.10 Griffin

Griffin is a MOOSE-based radiation transport application jointly developed by Idaho National Laboratory and Argonne National Laboratory. Griffin includes a variety of deterministic radiation transport solvers for fixed-source, k-eigenvalue, adjoint, and subcritical multiplication, as well as transient solvers for point-kinetics, improved quasi-static, and spatial dynamics.

Interfaces

- Mole (planned): Griffin could provide burnup and transmutation rates to Mole.
- Pronghorn: Griffin provides power distributions to Pronghorn.

- SAM: Griffin provides power distributions to SAM.

2.11 Grizzly

Grizzly is a MOOSE-based code for modeling the performance of structural components in a reactor environment [6]. Grizzly can model a variety of systems in both light-water and advanced reactors. The aspect of Grizzly that is most relevant to molten salt is its ability to model the mechanical response of alloys under high-temperature conditions, when creep is a significant factor. The thinning of structural component walls because of corrosion can be directly accounted for in these models. Future development is also planned to account for the effects of salt/radiation exposure on the material response.

Interfaces

- Yellowjacket (planned): Yellowjacket could calculate the wall thinning and species concentrations used in constitutive models.
- Griffin (planned): Griffin could provide radiation fluences for use in constitutive models that account for irradiation effects.

3. INTEGRATION PLAN

After meeting with representatives of the development teams working on each of these aforementioned efforts, changes were made to the following areas to foster the delivery of an integrated set of codes for solving the coupled set of physics models that affect the chemistry solution.

3.1 MSTDB-TP

No major changes to the scope or direction related to this effort are needed at this time.

3.2 MSTDB-TC

No major changes to the scope or direction related to this effort are needed at this time.

3.3 Saline

No major changes to the scope or direction are needed at this time. However, the following minor change should be considered:

- Rather than presenting it as a separate product, this API could be provided as part of the MSTDB-TP project to minimize the number of products supported by the NEAMS program.

3.4 Mole

- The following changes should be made:
  - For system-level MSR chemistry/corrosion modeling, the advection/diffusion of isotopes should be managed by the applicable TH codes (e.g., SAM for 1D modeling) rather than implementing models for this internally in Mole.
  - To facilitate code reuse of species tracking capabilities by 2D/3D tools, Mole should be restructured so that the capabilities for chemical reactions and species tracking utilize the models in MOOSE-CR to the greatest extent possible.
  - Empirical models for corrosion rates were developed for early demonstrations. These models should be replaced by models informed by computed equilibrium states and chemical-potential-driven transport, making use of system-level species conservation models.

3.5 Pronghorn

- The following changes should be made:
  - Pronghorn has a mass transport modeling capability, largely provided by the MOOSE modules. Going forward, it should be used in the NEAMS program for intermediate-fidelity, component-level 2D/3D corrosion modeling.
  - Efforts should be initiated to develop a low-fidelity, system-level corrosion modeling capability that is based on Pronghorn and SAM and couples the solutions to low-fidelity TH and mass transport equations with neutron conservation and depletion. This will provide a pathway for introducing an intermediate-fidelity, 2D/3D system-level code that can model corrosion and deposition and accounts for elemental composition changes due to nuclear processes (e.g., fission, transmutation, and decay).

3.6 MOSCATO

- The following change should be made:
  - MOSCATO's demonstrated ability to solve coupled TH, chemistry, and corrosion models for full loops should not be this code's primary function within the NEAMS program. Instead, this type of system-level computation should be performed using Mole, SAM, and Pronghorn. MOSCATO's primary role should be to provide correlations and closure models for mass transfer at complex geometric features to inform coarse, system-level TH models based on SAM and Pronghorn.

3.7 SAM

- The following change should be made:
  - SAM already has a mass transport modeling capability and should be adopted to provide mass transport for low-fidelity, system-level 1D corrosion modeling.

3.8 Corrosion Modeling via Yellowjacket and the MOOSE PhaseField Module

- The following change should be made:
  - All the capabilities for phase-field modeling of corrosion are provided by the MOOSE PhaseField (MOOSE-PF) module. The original intent was for Yellowjacket to contain application-specific phase-field code. However, thanks to the increased generality of MOOSE-PF, there is currently no apparent need for that. To date, all phase-field modeling of corrosion at the grain scale has been sufficiently handled by MOOSE-PF, and this trend is anticipated to continue for some time. As a result, these models are referred to as being based on MOOSE-PF and/or MOOSE-CR, rather than on Yellowjacket.

3.9 Thermochimica Gibbs Energy Minimizer

- The following change should be made:
  - Efforts will be shifted from developing the GEM solver in Yellowjacket to building an interface between MOOSE and Thermochimica for Gibbs energy minimization. To facilitate widespread usage of that interface by all applications, it will be located in an easily accessible product (most likely MOOSE-CR). This change is being made because, in the time since development of Yellowjacket began, Thermochimica was publicly released as open-source software and has thus enjoyed more widespread adoption and support than it had previously. Thus, there is less of a need for a MOOSE-native GEM, and consolidating efforts around Thermochimica will minimize

duplication of effort. Some design aspects of the GEM solver in Yellowjacket afford performance benefits for use within MOOSE; therefore, throughout the development of the Thermochimica interface in MOOSE, benchmarking will be performed to assess the relative performance of the two codes and help direct future development efforts.

3.10 Griffin

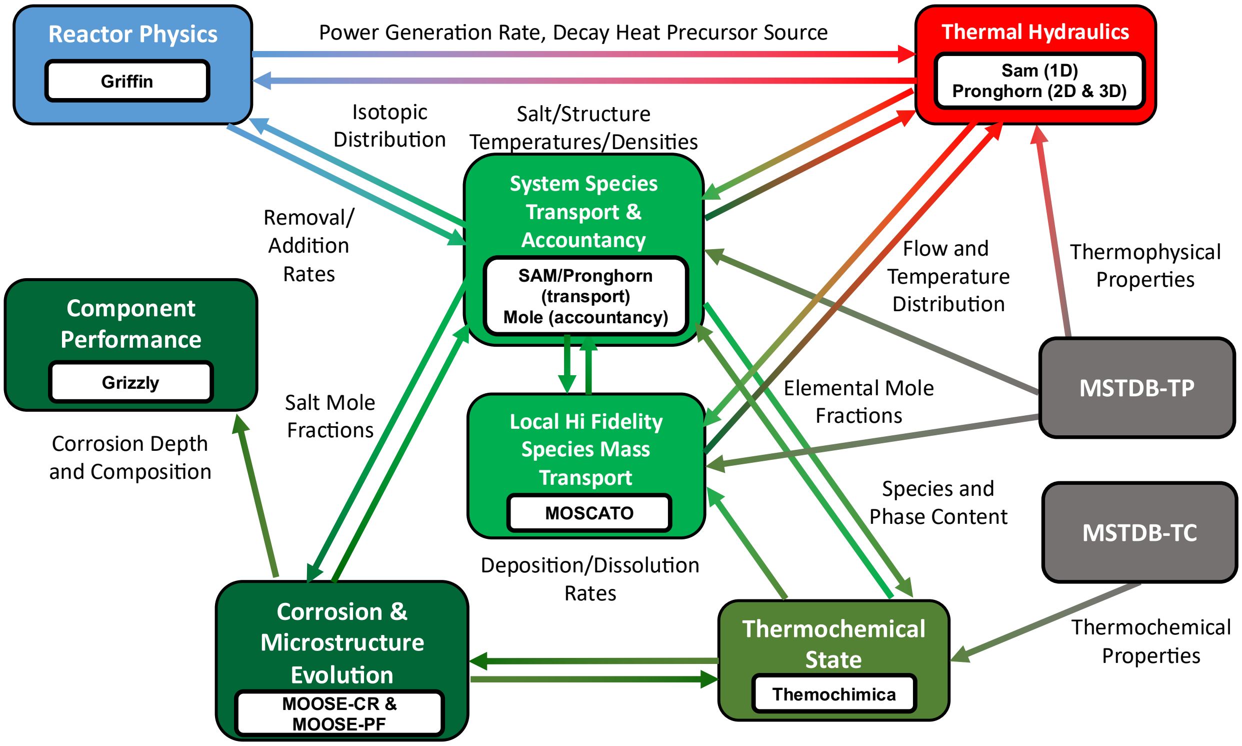
No major changes to the scope or direction related to this effort are needed at this time.

3.11 Grizzly

No major changes to the scope or direction related to this effort are needed at this time.

3.12 Revised Diagram of Physics and Code Relationships

Taking into account the above-described updates to the roles of the codes, Figure 2 shows a revised diagram of the MSR physical phenomena and their dependencies, along with the codes to be used in modeling each of those phenomena.



Reactor Physics
Griffin
Power Generation Rate, Decay Heat Precursor Source
Thermal Hydraulics
Sam (1D)
Pronghorn (2D & 3D)
Isotopic Distribution
Salt/Structure Temperatures/Densities
System Species Transport & Accountancy
SAM/Pronghorn (transport)
Mole (accountancy)
Removal/ Addition Rates
Component Performance
Grizzly
Flow and Temperature Distribution
Thermophysical Properties
Salt Mole Fractions
Local Hi Fidelity Species Mass Transport
MOSCATO
Elemental Mole Fractions
MSTDB-TP
Corrosion Depth and Composition
Species and Phase Content
MSTDB-TC
Deposition/Dissolution Rates
Corrosion & Microstructure Evolution
MOOSE-CR & MOOSE-PF
Thermochemical State
Themochimica
Thermochemical Properties

Figure 2. Proposed partitioning of physical phenomena and NEAMS-supported, MSR-focused codes and their interrelationships.

4. CONCLUSIONS

A suite of codes and databases is currently being developed by the NEAMS program to simulate key aspects of chemistry and corrosion in MSR systems. The current state of these codes, along with the relationships between those codes has been summarized in Section 2. To facilitate better interoperability of these codes and more efficient usage of resources, several changes, outlined in Section 3 are being made to these codes. Because of the complexity of this problem, as these codes are further developed and become more deeply integrated one with another, it is expected that this plan will need to be periodically reevaluated.

5. REFERENCES

1. Mcmurray, J., et al. 2021. "Integration roadmap for multi-scale, multi-physics mass accountancy in Molten Salt Reactors." ORNL/TM-2021/1866, Oak Ridge National Laboratory. https://doi.org/10.2172/1827046.
2. Ard, J. C., et al. "Development of the *Molten Salt Thermal Properties Database - Thermochemical (MSTDB-TC)*, example applications, and LiCl-RbCl and UF3-UF4 system assessments." *J. Nucl. Mater.* 563, 2022, 153631, ISSN 0022-3115. https://doi.org/10.1016/j.jnucmat.2022.153631.
3. Otis, R. and Z.-K. Liu, 2017. "pycalphad: CALPHAD-based Computational Thermodynamics in Python." *J. Open Res. Softw.*, vol. 5, no. 1, pp. 1. http://doi.org/10.5334/jors.140.
4. Yuan, H. and N. Hoyt. 2021. "MOSCATO Solver Development and Integration Plan." ANL/NSE-21/66, Argonne National Laboratory. https://www.osti.gov/biblio/1841593.
5. Bhave, C. V., M. R. Tonks, G. Zheng, K. Sridharan, "An electrochemical mesoscale tool for modeling the corrosion of structural alloys by molten salt." *J. Nucl. Mater.* 574, 2023, 154147, ISSN 0022-3115. https://doi.org/10.1016/j.jnucmat.2022.154147.
6. Spencer, B. W., W. M. Hoffman, S. Biswas, W. Jiang, A. Giorla, M. A. Backman. 2021, "Grizzly and BlackBear: Structural Component Aging Simulation Codes." *Nucl. Technol.* 207, no. 7, pp. 981–1003, (Apr.). https://doi.org/10.1080/00295450.2020.1868278.