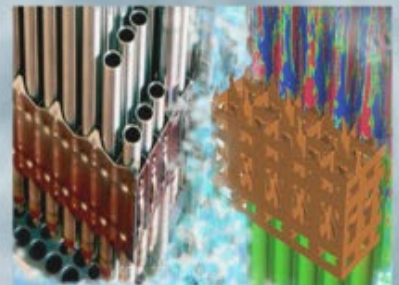
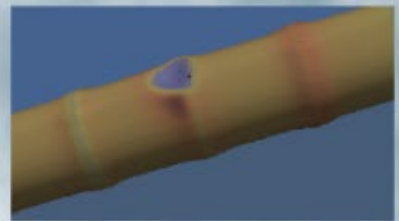
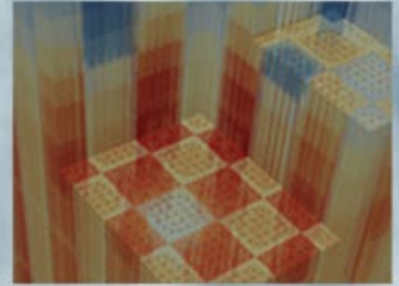


VERA 4.1 Release Notes

Mark Baird, ORNL
Ben Collins, ORNL
Will Cramer, ORNL
Andrew Godfrey, ORNL
Brendan Kochunas, UM
Ron Lee, ORNL
Rob Lefebvre, ORNL
Lori Moore, ORNL
Tara Pandya, ORNL
Bob Salko, ORNL
Shane Stimpson, ORNL
Erik Walker, ORNL

April 9, 2020



DOCUMENT AVAILABILITY

Reports produced after January 1, 1996, are generally available free via US Department of Energy (DOE) SciTech Connect.

Website <http://www.osti.gov/scitech/>

Reports produced before January 1, 1996, may be purchased by members of the public from the following source:

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
Telephone 703-605-6000 (1-800-553-6847)
TDD 703-487-4639
Fax 703-605-6900
E-mail info@ntis.gov
Website <http://www.ntis.gov/help/ordermethods.aspx>

Reports are available to DOE employees, DOE contractors, Energy Technology Data Exchange representatives, and International Nuclear Information System representatives from the following source:

Office of Scientific and Technical Information
PO Box 62
Oak Ridge, TN 37831
Telephone 865-576-8401
Fax 865-576-5728
E-mail reports@osti.gov
Website <http://www.osti.gov/contact.html>

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

REVISION LOG

Revision	Date	Affected Pages	Revision Description
0	4/9/2020	All	Initial Release

Document pages that are:

Export Controlled ___ None _____

IP/Proprietary/NDA Controlled ___ None _____

Sensitive Controlled ___ None _____

Requested Distribution:

To: Unlimited distribution

Copy:

1. VERA 4.1 SOFTWARE PRODUCT SUITE

The Virtual Environment for Reactor Applications (VERA) components included in this distribution include selected computational tools and supporting infrastructure that solve neutronics, thermal-hydraulics, fuel performance, excore radiation transport, and CRUD/chemistry problems for commercial pressurized water reactors (PWRs). In many cases, these tools can be executed stand-alone or coupled to other VERA components. VERA also provides a simplified common user input and output capability, and the infrastructure components support the physics integration with data transfer and coupled-physics iterative solution algorithms [1].

Neutronics analyses can be performed for 2D lattices, 2D cores, and 3D core problems for PWR geometries that can include reactor criticality and fission rate distributions at the fuel rod level for input fuel compositions. MPACT uses the Method of Characteristics (MOC) transport approach for 2D problems [2]. For 3D problems, MPACT uses the 2D/1D method which uses 2D MOC in each radial plane and 1D P_n in the axial direction. MPACT includes integrated cross section capabilities that provide problem-specific 51 energy group cross sections generated using the subgroup self-shielding methodology. MPACT also includes the isotopic depletion and decay capability in ORIGEN [3] for simulation of all reactor operating regimes as a function of time or burnup. MPACT also handles simulation of other aspects of reactor operation, such as control rod movement, fuel shuffling, refueling outage decay, and incore instrumentation response. The code can solve both 2D and 3D problems on parallel processors to significantly reduce overall run time.

A thermal-hydraulics capability is provided with CTF (an updated version of COBRA-TF) [4] that allows thermal-hydraulics sub-channel analyses for single and multiple assemblies using the simplified VERA common input. This distribution also includes coupled neutronics/thermal-hydraulics/depletion capabilities to allow calculations using MPACT coupled with CTF and ORIGEN, and version 4.1 also includes a new fuel rod temperature model in CTF that provides intra-pellet temperature coupling with MPACT for doppler feedback to the neutronics.

The VERA fuel rod performance component BISON [5] calculates, on a 1,5D, 2D, or 3D basis, fuel rod temperature, fuel rod internal pressure, free gas volume, clad integrity, and fuel rod waterside diameter. These capabilities allow simulation of power cycling, fuel conditioning and deconditioning, high burnup performance, power uprate scoping studies, and accident performance. In VERA, BISON is run uncoupled from the other components, but templates and tools are provided to automatically generate and execute BISON input for all fuel rods in a VERA core model (approximately 15,000 parallel BISON executions).

Note that BISON will NOT be included in VERA 4.1 distributions through RSICC. To obtain BISON, please contact jess.gehin@inl.gov. The corresponding version of the BISON software that has been tested with VERA 4.1 can be identified by INL with SHA key: 89b676d045cba9018c520afc5885a5778ae42169.

VERA 4.1 is distributed with the Shift Monte Carlo radiation transport code [6], which can be run stand-alone using a VERA common input with the `vera_shift` executable and directly coupled to MPACT/CTF using the `vera_to_shift` executable. Shift uses the SCALE ENDF/B-VII.1 continuous-energy data library and has been used to model a broad set of problems for CASL including massively parallel eigenvalue problems as reference solutions and a variety of coupled ex-core transport applications. The breadth of commercial reactor applications using VERAShift is partially documented in the Verification and Validation report shown in Table 1. VERAShift has recently been demonstrated to produce accurate ex-core detector responses and validation for vessel fluence applications is ongoing.

VERA 4.1 also includes the MAMBA [7] code for CRUD/chemistry capabilities. MAMBA provides CRUD growth and soluble boron uptake on the fuel rod surfaces based on the surface heat flux from MPACT and sub-cooled boiling duty from CTF. MAMBA also includes capability for system mass balance of the CRUD sources and CRUD removal and shuffling during refueling outages. A new CTF-informed sub-channel methodology implemented in CTF provides capability for producing very detailed local CRUD deposits with consideration for spacer grid heat transfer effects. MAMBA provides CIPS/CILC analysis capability, but is still maturing and requires more testing, calibration, and validation.

Input processing capabilities include the VERA Common Input (VERAIn) processor which provides a simple and user-friendly input interface for all VERA components [8]. VERAIn converts the ASCII common input file to an intermediate XML format which serves as a common geometric database for each of the physics components in VERA. VERA component codes either read the VERA XML format directly or provide a preprocessor which can convert the XML into native input.

VERAView [9] is an interactive graphical interface for the visualization and engineering analyses of output data from VERA. The python-based software is easy to install and intuitive to use, and provides instantaneous 2D and 3D images, 1D plots, and alpha-numeric data from VERA multi-physics simulations. VERAView is open source and can be obtained from GitHub at <https://github.com/CASL/VERAview>.

VERA 4.1 has been tested and applied broadly within CASL in FY19 and FY20 for milestone activities in the AMA focus area. In general, the following commercial power plants have been successfully simulated or analyzed with VERA 4.1, to some degree:

- Watts Bar Units 1 and 2
- Catawba Unit 1
- Oconee Unit 3
- Vogtle Units 1 and 2
- V.C. Summer Unit 1
- Three Mile Island Unit 1
- Byron Unit 1
- Davis-Besse Unit 1
- Sanmen Units 1 and 2
- Haiyang Units 1 and 2
- Krsko Unit 1
- NuScale SMR

VERA 4.1 has also been successfully utilized to generate solutions to the CASL Core Physics Benchmark Progression Problems [10], which is a publicly-available set of test problems based on Watts Bar Unit 1 that include both plant measurements and high-fidelity reference solutions for software testing and validation. The VERA 4.1 results to these problems can be provided upon request.

In FY19, Westinghouse performed a rigorous benchmark of eleven power plants (27 cycles) and evaluated VERA's performance for standard steady-state core follow analyses [11]. The components utilized for this work were MPACT, CTF, and ORIGEN. The results were excellent and are summarized below.

- The average hot-zero-power critical boron difference was 6 ppm.
- The average isothermal temperature coefficient difference was 0.3 pcm/°F.
- The average control bank worth error was 0% with a standard deviation of 4.1%.
- The average hot-full-power critical boron difference was 14 ppm at BOC and 27 ppm at EOC.
- The radial power distribution RMS was 1.5% and the total 3D RMS was 3.4%

2. SOFTWARE QUALITY ASSURANCE

The VERA 4.1 software product suite has been developed, validated, and tested under a new NQA-1 Quality Assurance (QA) Program established in 2019 [12]. This QA program initiative stemmed from nuclear industry feedback identifying the NQA-1 Quality Assurance standard used throughout the U.S. nuclear industry as a top priority to enable the removal of a key barrier for their adoption of VERA.

The VERA software components included in this distribution are still under active development and are subject to change. However, all software lifecycle work activities are performed, managed, and controlled under the NQA-1 compliant program. Software quality levels (SQLs) are established to define the controls, rigor, and formality applied to software engineering processes, documentation, verification, and validation activities based on the maturity of the code:

- **SQL1:** Mature core software product codes (CTF, MPACT, and VERAIO) are assigned SQL1 status to reflect the completion of specified product-specific management plans, input requirements and tests, baseline documentation, independent reviews, and other NQA-1 work activities.
- **SQL2:** Third Party Libraries (TPLs) and utilities that are used to support the SQL1 codes but do not directly implement the theoretical model are designated as SQL2. Acceptance for their use is documented in core product SMPs, baseline software documentation, and testing protocols.
- **SQL4:** Software that is in the research and development stage are included under the NQA-1 program as SQL4 to drive toward mature processes, controls and documentation consistent with SQL1. MAMBA and VERAShift are included in the distribution of VERA 4.1 as SQL4. **They have not been fully validated or assessed, and should be used primarily for test, evaluation, and research purposes only.**

The BISON software quality program is managed by Idaho National Laboratory (INL) separately from the other VERA components. BISON has recently completed a separate NQA-1 audit and is expected to be considered as SQL1 soon. For VERA 4.1, BISON should be considered for test and evaluation purposes only (SQL4).

Within the SQL1 codes (CTF, MPACT, VERAIO), there are both mature and research capabilities. The mature capabilities are rigorously implemented and tested prior to software release. The research capabilities are still under research and development and have not been thoroughly tested, demonstrated, or validated. These may not be supported features of VERA 4.1 but may be more mature in future releases. Examples are below.

- Examples of supported features: PWR steady-state core follow, depletion, decay, fuel shuffling, xenon transients, PWR stand-alone T/H analyses, single phase flow, etc.
- Examples of unsupported R&D features: BWR, fully-coupled transients (RIA), excore neutron transport calculations, CRUD analyses (CIPS & CILC), CFD-informed subchannel applications, etc.

During release testing of VERA 4.1, several defects and issues were identified and addressed by the code development teams. Some of these have been corrected in VERA 4.1 and some will be addressed in later VERA versions. Section 9 below contains a comprehensive list of known defects, issues, and workarounds in this release. Please report any additional new defects identified to support@casl.gov.

3. SYSTEM REQUIREMENTS

Linux platforms with functioning gcc, g++, and gfortran compilers and X11 libraries are supported. For quarter-core simulations of commercial reactor problems, with fidelity consistent with CASL analyses, a minimum of approximately 500 compute cores are needed with approximately 4 GB memory available per core. 1000 cores are recommended for these problems. When running VERA with VERAShift excore calculations, another 100-400 cores are recommended.

Detailed system software and third-party library requirements are specified in the provided VERA Installation Guide.

This distribution has been tested and verified to install and execute on the following OS distributions:

- CentOS 7
- CentOS 7.4
- RedHat 7.4
- SUSE 3.0.101
- SUSE 3.0.76
- CRAY OS
- Ubuntu 16.04.2

4. INSTALLATION

Detailed installation instructions are provided in the VERA Installation Guide located in the distribution tarball under the `VERA/doc/installation_guide` folder and with the documentation delivered with the distribution.

5. DOCUMENTATION

The following documentation is made available with the VERA 4.1 release distribution. Other CASL technical reports and publications are available at <http://www.casl.gov/journal-and-conference-papers.shtml> and <http://www.casl.gov/technical-reports.shtml>.

Table 1. VERA 4.1 Component Documentation

Document ID	Document Title
CASL-U-2019-1829-000	VERA 4.1 Installation Guide
CASL-U-2014-0014-004	VERA Common Input User Manual
CASL-U-2019-1886-000	CTF Theory Manual
CASL-U-2019-1885-000	CTF User's Manual
CASL-U-2019-1887-000	CTF Validation and Verification
CASL-U-2019-1874-001	MPACT Theory Manual
CASL-U-2018-1597-000	MPACT Standard Input User's Manual
CASL-U-2019-1873-001	MPACT VERA Input User's Manual
CASL-U-2019-1877-006	MPACT Verification and Validation Manual (Revision 6)
CASL-U-2017-1445-000 / LA-UR-17-29083	User Guidelines and Best Practices for CASL VUQ Analysis Using Dakota (https://permalink.lanl.gov/object/tr?what=info:lanl-repo/lareport/LA-UR-17-29083)
SAND2014-4253	DAKOTA 6.6 Theory Manual (https://dakota.sandia.gov/sites/default/files/docs/6.6/Theory-6.6.0.pdf)
SAND2014-4633	DAKOTA 6.6 User's Manual (https://dakota.sandia.gov/sites/default/files/docs/6.6/Users-6.6.0.pdf)
(Online)	DAKOTA Reference Manual (https://dakota.sandia.gov/content/latest-reference-manual)
(Online)	DAKOTA Developer's Manual (https://dakota.sandia.gov/content/latest-developers-manual)
INL/EXT-13-29930	BISON Theory Manual (https://bison.inl.gov/SiteAssets/BISON_Theory_ver_1_3.pdf)
INL/MIS-13-30307	BISON User's Manual (https://bison.inl.gov/SiteAssets/BISON_Users_ver_1_3.pdf)
CASL-U-2016-1099-001	File-Based One-Way BISON Coupling Through VERA: User's Manual
CASL-U-2017-1311-001	Excore Radiation Transport Modeling with VERA
CASL-U-2018-1556-002	Excore Radiation Transport Modeling with VERA: Manual
CASL-U-2016-1186-000	Shift Verification & Validation
CASL-U-2019-1836-000	MAMBA v.2.0 Theory Manual

6. SUPPORT

Questions, issues, bugs, and suggestions should be reported to support@casl.gov.

7. PHYSICS COMPONENTS INCLUDED IN VERA 4.1

This section describes the new, stable, and experimental features included in this release of VERA. For a full list of known defects and issues, please see Section 9.

The component code versions included with this distribution and, for reference, the previous VERA versions are listed in Table 2. Cryptographic labels refer to CASL Git repository SHA keys that uniquely identify code versions:

Table 2. VERA 3.7, 3.8, 4.0 and 4.1 Component Versions

Component	VERA 3.7 Version	VERA 3.9 Version	VERA 4.0 Version	VERA 4.1 Version
MPACT	e0216dd (Aug 9, 2017)	706de3d (Feb 17, 2018)	b5e2cbe (Mar 20, 2019)	869ab07 (Oct 17, 2019)
CTF	f5a9e62 (Aug 10, 2017)	9105f20 (Feb 13, 2018)	30aab7c (Mar 25, 2019)	97514a5 (Nov 14, 2019)
VERAIO	N/A	N/A	N/A	N/A
BISON	0cdf98c (Aug 14, 2017)	f293c4d (Jan 16, 2018)	be31f34 (Feb 26, 2019)	68913d5 (May 29, 2019)
VERAShift	f928902 (Aug 10, 2017)	911aa80 (Feb 14, 2018)	cc034ad (Mar 19, 2019)	ac561d8 (June 6, 2019)
MAMBA	N/A	N/A	d971461 (Mar 25, 2019)	6bd9ab2 (May 22, 2019)
Dakota	V6.6	V6.6	V6.6	V6.6

Complete version information for all CASL source code repositories associated with this release are documented in CASL-U-2020-1941-000 which is available from CASL Records.

7.1 MPACT

New features implemented since VERA 4.0 are:

- Various bug fixes

New features implemented since VERA 3.9 are:

- Various bug fixes
- Updated documentation
- In-line thermal expansion processing (ThermalExpandXML.exe preprocessor is no longer needed)
- Generation of summary output file
- New input format for “jump-in” cycles
- Support for accident tolerant fuel forms (U_3Si_2 and UN) and cladding (FeCrAl and SiC/SiC)
- Better usability and parallel performance of spatial partitioning via graph partitioning
- Linear Source method of characteristics solver
- Addition of ^{15}N to cross section library
- Nodal cross section edits
- Improved ASCII output formats
- Consistent check on core height
- Explicit delayed energy deposition for transients
- Intra-pin edits for VERAView
- Coupling of intra-pin power, temperature, and burnup with CTF

The following features are stable:

- Support for Linux OS (32-bit and 64-bit)
- Parallel Spatial Decomposition with MPI
- Parallel Angular Decomposition with MPI
- User defined Macroscopic Cross Sections
- 51-group Macroscopic Cross Section Library Data
- Subgroup Resonance Self-Shielding
- Transport Corrected P0 scattering treatment
- Export of Mesh to Legacy VTK and VTU file for visualization
- 2-D MOC Transport Kernel
- Coarse Mesh Finite Difference (CMFD) Acceleration
- 1-D Nodal Kernels based on NEM-Diffusion and SPn
- 2-D/1-D Full Core Solution
- Multi-State Calculation Capability
- Transient Calculation Capability
- Depletion and Decay
- Critical Boron Search
- Equilibrium Xenon Calculation
- Direct Coupling with COBRA-TF
- Simplified Internal T/H
- General PWR Geometry Modeling
 - IFBA
 - Control Rods and Control Rod Banks
 - Burnable Poison Inserts
 - Fission Chamber Detectors
 - Grid spacers, Nozzles, Plenum, Baffle, etc.
 - Semi-Explicit modeling of grid spacers
- Isotopic Restart File
- Cycle-to-Cycle Fuel Shuffling
- Radial Thermal Expansion

The following features are considered to be experimental:

- Separate B-10 Depletion of soluble Boron
- Space-Dependent Wielandt Shift or dynamic Wielandt Shifts
- Modeling of BWRs
- Simplified CRUD modeling
- Post-Corrector Depletion
- 3-D MOC Solvers
- Processing of AMPX Working Cross Section Libraries
- Processing of ISOTXS Cross Section Libraries

- Secondary Source Calculation
- Axial Thermal Expansion
- Multi-level in Space and Energy Diffusion Solver for CMFD
- Remeshing and fuel rod reconstitution in restart and shuffle
- Center assembly homogenization when shuffling fuel
- Coupling with MAMBA crud chemistry
- Coupling with Shift for ex-core detector response calculations
- Subplane Treatment

7.2 CTF

CTF (an updated version of the COBRA-TF code) is a subchannel thermal-hydraulics code that uses a two-fluid, three-field (i.e. fluid film, fluid drops, and vapor) modeling approach [4]. Both sub-channel and three-dimensional (3D) Cartesian forms of nine conservation equations are available for LWR modeling. CTF includes a wide range of thermal-hydraulic models important to LWR safety analysis and modeling of normal LWR operating conditions including flow-regime-dependent, two-phase wall heat transfer, inter-phase heat transfer and drag, turbulent mixing, void drift, droplet breakup, and quench-front tracking. Due to its 3D capabilities and extensive array of reactor thermal-hydraulic modeling capabilities, CTF has found much use in modeling of LWR rod-bundle transient analysis and Pressurized Water Reactor (PWR) whole-vessel, Loss-Of-Coolant Accident (LOCA) analysis.

Significant changes implemented since VERA 3.9 are:

- Add clad corrosion model to CTF
- Improved IO for crud simulations
- Improvements to ROTHCON for crud simulations
- Add new options for the source term model
- Change code defaults for better numerical stability
- Add clad creep model to CTFFuel
- Fixes to xml2ctf bug for generating multi-assembly models with large water rods for BWRs
- Switch steam table default to IAPWS97 lookup tables
- Switch SPARSKIT2 solver library out with Futility
- Remove MAMBA1D as a crud solver option in CTF
- Remove crud and source term model surrogates in CTF
- Add more models to CTFFuel for fuel swelling and clad hoop stress
- Add new parallel decomposition options to use 9 and 16 processors per assembly
- Shut off text file outputs from CTF to prevent large amounts of data being written for large files
- Expose HDF5 handle from CTF coupling interface so CTF and MPACT can write to the same HDF5 file
- Fix bug in transient CHF model
- Add preliminary non-condensable gas tracking model to the species transport class

- Switch to gcc 5.4.0
- Fix MAMBA restart procedures
- Removed deprecated CTF preprocessor (using text input files) from the repository
- Fix how multistate driver reads flow from the VERA xml input file
- Fix how CTF treats crud quantities used in the mass balance model for quarter symmetry models
- Improve the Multistate_cobra driver so it can take the VERA xml file as an input file (instead of the Multistate_cobra input file)
- Make the Multistate_cobra input file more consistent with the VERAIn file format
- Fix defect where xml2ctf would create an invalid CTF model for single pins when parallel is turned on
- Improvements to the pressure matrix solve for improved code performance in smaller models
- Modify CTF to recognize VERAIn repo during the configure if a user clones it into CTF
- Add a fluid property class that allows developers to specify temperature dependent custom defined fluid properties
- Change default turbulent mixing coefficient from 0.005 to 0.037, which was found to be more realistic for models with mixing vane grids
- Modify xml2ctf to use the coupling mesh created by MPACT and added to the VERA xml parameter list to build the CTF axial mesh instead of the axial edit bounds in VERAIn (axial edit bounds will be used if the coupling mesh is not available).
- Fix boundary conditions passed to MAMBA, which reduces the occurrence of NaNs appearing in the crud solution.
- Modify CTF to recognize MAMBA repo during configuration if the user clones it into CTF
- Fix Docker build script for updated TPL versions
- Add pin TKE map to the VERA HDF5 file
- Fix bug in averaging of crud boundary conditions passed to MAMBA for symmetry rods
- Add ability to model grids in smeared regions (inactive regions) in xml2ctf
- Expose gas species transport model to the coupling interface
- Add models for transport of species between phases
- Add restart capability for the CTF fuel rod model so that MPACT can include fuel rod restart data in the restart file
- Add setters for fast flux and irradiation time in the coupling interface for the CTF fuel rod model
- Add save/rewind capability to the CTF fuel rod model for modeling fuel in depletions
- Add depletion modeling capability to CTFFuel input
- Add more core TH data to the VERA HDF5 file
- Improve consistency of VERA HDF5 dataset naming and improve on documentation attributes
- Add additional CTF fuel rod data to the VERA HDF5 file
- Update the CTF code style guide
- Update CTFFuel model defaults to be more realistic and consistent with other fuel solvers

- Apply ROTHCON multipliers to the rod surface data printed to the HDF5 file (when ROTHCON is used)
- Add switch for Groeneveld CHF tables in VERAIn file
- Extend parallel CTF so it will work for multisection models
- Simplify the parallel data required in the CTF input file so users do not need to specify it for the gaps and only need to specify channel owner domains for channels (CTF will figure out what channels need to be ghosts).
- Improve parallel input error checks
- Add a new MPI data share algorithm for small models that is more efficient and offers better performance than the current one (CTF will set the algorithm automatically depending on model size).
- Get parallel CTF working for the species transport model
- Remove two-phase datasets from VTK file for salt simulations
- Add ability to write TH solution data to the debug file when that option is selected
- Add ability to set direct heat deposition in fuel rod clad, guide tube metal, and fluid on a channel-wise/pin-wise basis from MPACT.
- Add ability for MPACT to set radial burnup distribution shape in CTF pins on a pin-wise/level-wise basis.
- Change grid blockage ratio default from 0.0 to 0.5
- Change guide tube model coefficient (for simple guide tube heating model) so a temperature rise is captured in the guide tubes for TH feedback in MPACT.
- Add SubKit to the CTF repository
- Fix bug in grid spacer enhancement factor applied to TKE for crud simulations
- Add ROTHCON data to the VERA HDF5 file for use in VERAView
- Add support for restarting CTFFuel calculations
- Expose CTFFuel gap heat transfer coefficient under-relaxation coefficient to input file
- Add thermophysical property reader to CTF so the user can specify their own table-based fluid properties for the code to use
- Additional bug fixes

Stable features:

- Solid modeling capabilities
 - Radial conduction
 - Nuclear fuel rod models (pellet, gap, and clad regions and UO₂ and zircalloy material properties)
 - Dynamic gap conductance model (pellet relocation and pellet-clad interaction)
- Fluid modeling capabilities
 - Solid-to-fluid heat transfer
 - single-phase convection
 - subcooled/saturated boiling
 - Critical heat flux (Departure from Nucleate Boiling)

- Two-phase flow with droplets
- Closure models
 - Wall drag and form loss modeling
 - Turbulent-mixing and void-drift
 - Fluid equation of state
 - Droplet entrainment and de-entrainment
- Incorporation of PETSc solvers
- Variable size axial meshing
- Grid-heat-transfer enhancement modeling
- Multisection modeling with channel splitting and coalescing
- General species transport

The following features are experimental:

- Solid modeling capabilities
 - Fuel rod axial/azimuthal conduction
 - Axial mesh refinement (quench front tracking)
 - Zircalloy-water thermal reaction
- Fluid modeling capabilities
 - Non-condensable gas effects
 - Post-CHF heat transfer models (are encountered in validation tests, but no validation of models done)
 - Channel flow area variations (rod ballooning)
 - Grid-directed cross-flow modeling
 - Boron-tracking model with consideration of boron precipitation

7.3 BISON

Note that BISON will NOT be included in VERA 4.1 distributions through RSICC. To obtain BISON, please contact jess.gehin@inl.gov. The corresponding version of the BISON software that has been tested with VERA 4.1 can be identified by INL with SHA key: 89b676d045cba9018c520afc5885a5778ae42169.

VERA includes the capability to predict fuel rod performance utilizing 2D axisymmetric or 3D coupled multi-physics and represents a significant advancement for the modeling/analysis capabilities in LWR fuel rod behavior [5]. The capability is being constructed within the MOOSE/BISON computational framework (from INL) that supports:

- 1.5D-RZ, 2D, and 3D thermo-mechanics including elasticity, plasticity with strain hardening, creep, large strains, large displacements, and smeared plus explicit cracking;
- Unsteady (transient) conduction heat transfer with time and spatial (axially, radially and potentially azimuthally in a cylindrical fuel element) dependent internal heat generation;
- Gap heat transfer including conduction, radiation and enhanced heat transfer from mechanical contact;

- 2D axisymmetric, generalized plane strain, and plane stress representations, including thermal and mechanical contact interactions between pellets and between the pellet and cladding;
- Mixed dimensional coupling, e.g., combined 2D and 3D numerical representations for coupled global (2D) and local effects (3D) modeling; and
- Utilizes high performance computing platforms to achieve the parallel performance and scalability required to perform coupled multi-physics simulations of full length 3D representations of the fuel rod components.

The BISON fuel rod performance code architecture uses the finite element method for geometric representation and a Jacobian-free, Newton-Krylov (JFNK) scheme to solve systems of partial differential equations. The fuel rod performance capability includes models for:

- Clad stress, strain, and strain rate;
- Clad oxidation, hydrogen pickup and hydride formation;
- Pellet stress, strain, and strain rate;
- Fission gas release (transient and pseudo-steady-state);
- Pellet densification and swelling;
- Pellet cracking (isotropic and smeared) and relocation;
- Thermal expansion, including pellet hour-glassing;
- Thermal and irradiation creep;
- Thermal conductivity effects due to clad oxidation;
- Material strength and ductility effects due to irradiation;
- Pellet-cladding gap evolution and local stress due to partial contact;
- Pellet stack growth and fuel rod growth;
- Explicit modeling of duplex and triplex clad designs;
- Reference residual calculations for improved robustness;

The VERA fuel rod performance subcomponent calculates, on a 2D or 3D basis, fuel rod temperature, fuel rod internal pressure, free gas volume, clad integrity and fuel rod waterside diameter. These capabilities allow simulation of power cycling, fuel conditioning and deconditioning, high burnup performance, power uprate scoping studies, and accident performance.

It is important to note that these tools are principally built around the known performance of existing zirconium-based clad with UO_2 fuel. Estimates for the global effects of minor modifications to the fuel or clad may be possible; for example, chromia-doped pellets may be simulated with user-supplied models for several of the pellet performance characteristics or steel-based clad may be simulated with similar user-supplied models. Materials such as silicon carbides that do not fit the system paradigm can be simulated but are not likely to provide accurate results.

7.4 VERAShift

VERAShift is a Monte Carlo radiation transport framework. It was first distributed as part of VERA 3.7, and this release builds upon the ex-core capabilities added in VERA 3.9 and VERA 4.0. As before, VERAShift can be run standalone in eigenvalue mode using the VERA common input with

the `vera_shift` executable. It can also run coupled to the core simulator using the `vera_to_shift` executable, which as before, runs MPACT/CTF and Shift on different processor domains.

Changes since VERA 4.0 include:

- Updated user manual: CASL-U-2018-1556-001 titled “Excore Radiation Transport Modeling with VERA: Manual”
- General bug fixes
- Performance improvements including reduced memory usage
- Improved ex-core modeling
- Ability to use fission source spectrum from MPACT for Shift calculations
- Output of vessel fluence and detector response by MPACT in the standard HDF5 output file and restart files (vessel fluence can be visualized in VERAView)
- Output of adjoint flux from CADIS calculations on a pin-by-pin basis

Stable features of VERAShift in VERA 4.1:

- node-based parallelism using domain replication
- multistate calculations
- eigenvalue, forward only, and CADIS modes
- fission source coupling between VERA and Shift
- pincell isotopic coupling between VERA and Shift
- flux tallying in core barrel, core pads, vessel liner, and vessel using VERA common input
- supplemental geometry and tally input for ex-core model features and tallies (extra geometric features that cannot be modeled with VERA common input)

The following features are available but are considered experimental as they have not been tested rigorously:

- Transfer of temperatures and densities in fuel, clad, coolant of pincells and inserts between VERA and Shift
- Automated generation of ex-core model with bioshield and detectors from VERA input

Known limitations include:

- Memory usage when using unique pins is very high
- Memory usage for full core calculations is high
- Memory usage when using the full energy spectrum for the fission source from MPACT is high
- Memory usage when transferring fission source, isotopics, temperatures and densities from VERA is very high (requires unique pins)
- When running in CADIS mode, the parallel decomposition used for the deterministic adjoint calculation dictates the number of processors the VERAShift calculation must use

7.5 MAMBA

The MAMBA package [7] simulates growth of crud, which refers to metal oxide corrosion products (primarily nickel ferrite, NiFe_2O_4) on fuel cladding and accumulation of boron in the porous crud. Precipitation of boron compounds, such as lithium tetraborate ($\text{Li}_2\text{B}_4\text{O}_7$), can lead to a crud-induced power shift (CIPS) in the nuclear fuel. In addition, the crud itself can lead to crud-induced localized corrosion (CILC) due to reduced thermal transport and thus increased temperatures, which can cause mechanical failure of the fuel.

The role of MAMBA is to simulate the buildup of crud and the precipitation of boron rich compounds within the porous crud layer. Since the formation of crud is a fundamentally multiphysics problem, MAMBA is coupled to neutronic and thermal hydraulic solvers present in VERA to predict CIPS. MAMBA requires thermal hydraulics conditions as input, in particular the cladding surface heat flux, turbulent kinetic energy and the coolant temperature. Within VERA thermal hydraulic conditions are provided by CTF. Additionally, the crud source term is modeled in MAMBA, which originates from corrosion of steam generators and primary loop piping.

MAMBA is available in VERA 4.1 but all features are considered experimental at this time.

7.6 Dakota

The Dakota 6.6 package [13] manages and analyzes ensembles of simulations to provide broader and deeper perspective for analysts and decision makers. In its simplest mode, Dakota can automate typical parameter variation studies through a generic interface to a physics-based computational model. This can lend efficiency and rigor to manual parameter perturbation studies already being conducted by analysts. Dakota also delivers advanced parametric analysis techniques enabling design exploration, optimization, model calibration, risk analysis, and quantification of margins and uncertainty with such models. It directly supports verification and validation activities. Dakota algorithms enrich complex science and engineering models, enabling an analyst to answer crucial questions of:

- Sensitivity: Which are the most important input factors or parameters entering the simulation, and how do they influence key outputs?
- Uncertainty: What is the uncertainty or variability in simulation output, given uncertainties in input parameters? How safe, reliable, robust, or variable is my system? (Quantification of margins and uncertainty, QMU)
- Optimization: What parameter values yield the best performing design or operating conditions, given constraints?
- Calibration: What models and/or parameters best match experimental data?

An updated CASL Technical Report providing user guidelines and best practices for CASL VUQ analysis using Dakota (CASL-U-2016-1233-000/SANDIA Report SAND2016-1161) is available at: <https://dakota.sandia.gov/sites/default/files/documents/SAND-CaslDakotaManual-2016.pdf>.

The following features are considered to be mature and robust:

- Parameter studies: list, vector, centered, multi-dimensional

- Uncertainty quantification: Monte Carlo and Latin hypercube sampling, local reliability (probability of failure) methods, stochastic expansions (polynomial chaos and stochastic collocation)
- Optimization/calibration: gradient-based local, derivative-free local (pattern search), global (genetic algorithms, direct, etc.), local least squares, surrogate-based local methods
- Surrogate models: polynomials, Gaussian process/Kriging, neural network
- Parameter types: all are mature except discrete string and categorical types
- Interfaces: system, fork, and direct
- Responses: objective functions, calibration terms (including experimental data), and response functions

The following features are stable:

- Design and analysis of computer experiments: DDACE grid, random, orthogonal array, OA-LHS; FSU quasi-Monte Carlo (Halton, Hammersley, centroidal Voronoi tessellation), PSUADE Morris one-at-a-time
- Uncertainty quantification: global reliability (probability of failure) methods, adaptive stochastic expansions, importance sampling (including adaptive, and surrogate-based), Probability of Failure Darts, epistemic interval uncertainty, Dempster-Shafer, Bayesian inference (QUESO, DREAM), incremental LHS
- Optimization: NOMAD directional search, surrogate-based global including EGO, hybrid and pareto optimization
- Surrogate models: MARS, Taylor/TANA, hierarchical and multi-fidelity
- Interfaces: Python interfaces; work directory and parallel interface scheduling features refactored recently

The following features are experimental:

- Design and analysis of computer experiments: DDACE Box-Behnken, central composite designs
- Uncertainty Quantification: D-optimal sampling, Multi-level and multi-fidelity Monte Carlo methods, topology-based adaptive sampling, Bayesian inference (GPMSA)
- Optimization: Genie Opt-Darts, Genie Direct
- Surrogate models: moving least squares, radial basis functions, active subspace methods for dimension reduction
- Variables: string/categorical variable support is limited
- Interfaces: Matlab, Scilab and grid
- Responses: field data, for both simulations and experiments, including interpolation capability

Release notes for Dakota 6.6 (and previous versions whose capabilities are also included in Dakota 6.6) are available at <https://dakota.sandia.gov/content/release-notes>. Known Limitations of Dakota 6.6 are listed in Section 9.

8. VERA INPUT/OUTPUT TOOLS

8.1 VERAIIn

The VERA Common Input (VERAIIn) [8] provides a simple and user-friendly input interface for the VERA components. It is a PERL script which converts the ASCII common input file to the intermediate XML format used as an input and geometric database for the physics codes in VERA. VERA component codes either read the VERA XML format directly or provide a preprocessor which can convert the XML into native input (CTF and BISON). When processing the ASCII common input file, VERAIIn performs basic error checking to ensure that common input errors are caught before running a simulation. Additionally, VERAIIn performs geometry processing by expanding symmetric geometry inputs to their full geometry.

8.2 VERAView

VERAView is an interactive graphical interface for the visualization and engineering analyses of output data from VERA [9]. The python-based software is easy to install and intuitive to use, and provides instantaneous 2D and 3D images, 1D plots, and alpha-numeric data from VERA multi-physics simulations. It reads files in HDF5 format that meet the VERAOut specification [14]. A user's manual is provided giving a brief overview of the software and descriptions of the major features of the application, including examples of each of the encapsulated 'widgets' that have been implemented thus far.

VERAView currently requires Python-2.7 and the following packages:

Table 3 VERAView Required Python Packages

Package	Minimum Version
h5py	2.9.0
hdf5	1.10.4
matplotlib	1.5.1
mayavi	4.5.0
numpy	1.11.3
pillow	3.0.0
pyparsing	2.4.2
scipy	1.2.1
wxpython	3.0

VERAView is packaged a prebuilt Python environment based on Miniconda (<https://docs.conda.io/en/latest/miniconda.html>) and including all required packages. Single-click GUI installers are provided for Windows and Mac OSX, and an installation script is provided for Linux (based on RedHat Enterprise Linux / CentOS version 6).

VERAView interprets the category or type of datasets in the VERAOut based on their shape and size, and the type determines which widgets can be used to display the data. The following dimensions are determined from the datasets:

- NASS – Number of fuel assemblies in the calculated geometry (quarter or full core)
- NAX – Number of axial planes in the core region (assume same for all data)
- NPIN – Number of fuel rods across a fuel assembly (assumes equal X and Y dimensions)
- NCHAN – Number of coolant channels across an assembly (assumes equal X and Y dimensions)
- NDET – Number of in-core instrument strings
- NDETAX – Number of detector axial planes
- NFDETAX – Number of fixed detector axial planes
- NR – Number of fluence radial bins
- NTHETA – Number of fluence angle bins
- NZ – Number of fluence axial planes

Datasets for individual statepoints are stored in the HDF5 file in groups with name `/STATE_ nnnn`, where `nnnn` starts at 0001 and increments thereafter. Dataset types that are recognized are summarized in the following tables.

Table 4 Primary VERAVIEW Dataset Types

Type	Shape	Description
channel	(npiny+1, npinx+1, nax, nass)	3D coolant channel data
detector	(ndetax, ndet)	3D detector signals
fixed_detector	(nfdetax, ndet)	3D fixed detector signals
fluence	(nz, ntheta, nr)	3D fluence data
node	(1, 4, nax, nass)	nodal data
pin	(npiny, npinx, nax, nass)	3D fuel rod data
radial_detector	(ndet,)	axially-integrated (radial) detector distributions
scalar	()	scalar quantity

Derived datasets representing the average, root mean square, standard deviation, and sum across data axes are shown in the following table. VERAVIEW can calculate these datasets if they are not already in the VERAOut file.

Table 5 Derived VERAVIEW Dataset Types

Type	Shape	Description
:assembly	(nax, nass)	axially-integrated (radial) assembly-wise distributions
:axial	(nax)	radially-integrated (axial) distributions
:chan_radial	(npiny+1, npinx+1, nass)	axially-integrated coolant channel data
:core	()	core-wise values
:radial	(npiny, npinx, nass)	axially-integrated distributions
:radial_assembly	(nass)	axially- and radially- integrated distributions

Multiple files may be opened and viewed simultaneously provided they have congruent core geometries.

VERAVIEW provides several “basic capability” and “extended capability” or experimental widgets. Whereas the former are covered in the VERAVIEW Software Test Plan Requirement and Test

Report, experimental widgets are lightly tested. In addition, VERAView is designed to be extensible, support addition of custom widgets. Refer to the VERAView API and Programmer's Guide. [15]

Table 6 VERAView Basic Capability Widgets

Widget	Display	Datasets Supported
Core 2D View	Assemblies in the current geometry	channel, pin, :assembly, :chan_radial, :node, :radial, :radial_assembly
Core Axial 2D View	Vertical cut along assembly column or row	channel, pin, :assembly, :node
Assembly 2D View	Lattice view of selected assembly	channel, pin, :chan_radial, :radial
Axial Plots	Plots with axial level as the Y-axis	any with an axial dimension
Time Plots	Plots with time as the X-axis	channel, detector, fixed_detector, fluence, pin, radial_detector, scalar, :assembly, :axial, :chan_radial, :core, :node, :radial, :radial_assembly

Table 7 VERAView Experimental Widgets

Widget	Display	Datasets Supported
Detector Multi View	Plots and numerical display of detector values	detector, fixed_detector, radial_detector
Table View	Tabular view of dataset values at current selections	channel, detector, fixed_detector, fluence, pin, radial_detector, scalar, :assembly, :axial, :chan_radial, :core, :node, :radial, :radial_assembly
Vessel Core 2D View	Vessel fluence horizontal slice	fluence
Vessel Core Axial 2D View	Vessel fluence vertical slice	fluence
Volume Slicer 3D View	Cuts across the three dimensions	pin, :assembly, :radial
Volume 3D View	Volumetric view with cuts	pin

VERAView is no longer included in the VERA distribution package. It is available on GitHub at <https://github.com/CASL/VERAview>. The latest development version is also available for use by contacting support@casl.gov.

8.3 VERARun

VERARun is a driver utility designed to automate job submission on a particular computing platform. VERARun evaluates the user input and is configured for each machine specifically such that it can automatically create the scripts needed by a queuing system and easily submit the job for a user with only one simple command line execution. It is provided as Python package source distribution which must be extracted and installed using VERA's Python 2.7 environment.

After installing, two executable scripts will be installed in the Python environment: `verarun` and `verastat`. The former submits jobs, and the latter displays a history of submitted jobs. All command-line options and processing features are described by requesting usage help from the command line:

```
verarun -help
```

In its simplest form, `verarun` can be executed with the name of the VERAIn file, e.g.,

```
verarun 5a_2d.inp
```

For more information on VERARun please contact support@casl.gov.

9. CAVEATS AND KNOWN ISSUES

The VERA software components provided in this release are still under active development and may be subject to rapid change. As described in Section 2, the CASL QA program has established software quality levels to represent the stages of maturity exhibited by each code. SQL1 codes (MPACT, CTF, VERAIO) have been rigorously developed and tested using a successfully audited NQA-1 compliant QA program. SQL4 codes (VERAShift and MAMBA), however, are less developed, less tested, and still under significant active research. They should be used for test, evaluation, and research purposes only.

Issues that have been identified during development and testing of this release package are listed below. Some of these issues are software errors (reported as part of the software problem reporting process), and some are minor issues or nuances that have been encountered in the past and have not been resolved. New issues that are identified that are not listed here should be reported to support@casl.gov as soon as possible. Software problem reports for defects found in VERA are posted online at <https://www.casl.gov/corrective-actions>.

Table 8. Known Issues and Defects in VERA 4.1

Ticket	Component	Issue
5414	MPACT	For some MPI distributions the communication routines for the MG MOC kernels will experience an error in MPI. * WORKAROUND: Use the <code>shield_moc_kernel 1g</code> or <code>moc_kernel 1g</code> option
3434	MPACT	On some platforms, depletion cases run with spatial and angular decomposition will fail. * WORKAROUND: Use threading instead of angle decomposition.
3451	MPACT	When using TCP0, in some cases can drive the solution negative, and this can lead to cases diverging. This is more likely to happen in 3D problems with large reflectors * WORKAROUND: Use the <code>moc_kernel mg</code> option. Using P2 scattering is also an option at the expense of longer run times.
N/A	MPACT	The number of azimuthal divisions in the Fine Source Region mesh of a fuel pin or guide tube in the visualization file is not representative of the computational mesh. The visualization contains extra divisions to approximate a curved surface as a series of line segments.
5414	MPACT	For some MPI distributions the communication routines for the 1G MOC kernels will experience a memory leak in MPI. * WORKAROUND: use <code>moc_kernel mg</code> and <code>moc_mg_data_passing true</code>
5875	MPACT	The restart and shuffle files can use compression features of HDF5 (off by default). On occasion, it has been observed that decompression of the file leads to heap fragmentation. * WORKAROUND: disable writing the file with compression using <code>rst_compress 5</code> (default) * WORKAROUND: Use the h5repack utility to decompress the file before running MPACT. <code>h5repack -f NONE <compressed.res> <uncompressed.res></code>
N/A	MPACT	Support for the depletion of absorbing materials in control rods is not yet implemented. The absorber material in a control rod should be defined in the <code>CONTROL</code> block to ensure rod materials are not depleted. If the materials are placed elsewhere, they may be flagged as depletable.
5894	MPACT	Entries in the <code>shuffle_label</code> and <code>assm_map</code> must be consistent geometrically between fuel cycles. Shuffling by assembly serial number is not implemented. Shuffling errors don't produce useful messages
6147	MPACT	Assemblies with stainless steel rods must have the same thermal expansion temperatures with performing core shuffles
5644	MPACT	Center assembly homogenization during core shuffles does not work when changing the axial mesh
6134	MPACT	Statepoint summary in *.sum file has ***** for core flow rate
5771	MPACT	New jump-in input format does not work with quarter symmetry

6011	MPACT	Occasionally ORIGEN returns nan for some problems
6044	MPACT	Calculated boron concentration not retained between states when control rods are moved
6081	MPACT	Incorrect dataset shapes in HDF5 for BWR cases
6243	MPACT	Error when using ZAID 40000 in cladding material
5493	MPACT	Full core cases (steady-state and RIA) require increased memory availability compared to previous versions of VERA, due partially to the improvements in CTFuel. Memory requirements have been observed as high as 5.2 GB per process.
5989	MPACT	Geometry error if many digits are specified on the axial_edit_bounds
6019	MPACT	Small axial meshes created when edit mesh doesn't match fuel boundaries
6021 5929 6064	MPACT	'End of file' Error occasionally during thermal expansion
6280	MPACT	Small axial plane created unintentionally in upper reflector region
4563	MPACT	Axial thermal expansion is not recommended or supported
6002	MPACT	Error when using AIC control rods and FeCrAl guide tubes
6007	MPACT	Segmentation fault when BISON/shuffle_xml is empty
6159	MPACT	Direct moderator heating fraction in MPACT does not match CTF or user input value for calculation of fuel temperatures when input temperature tables as used
6274	MPACT	pin_fuel_temp output dataset is incorrect when using old fuel temperature table input format
6234	MPACT	Extended coupling mesh TH passed back from CTF not placed in MPACT properly
5951	MPACT	Axial remesh between fuel cycles seg faults if there is a change in the upper reflector
6015	MPACT	Axial mesh boundaries must align with spacer grid boundaries in the active fuel region
6017	MPACT	Smaller timesteps are required for depletion of fuel bearing the Gadolinia neutron absorber
6255	MPACT	Segfault encountered when running 3D external source driven problems
5629	MPACT	Non-standard diffusion coefficients used by default in external source calculations
5995	MPACT	Custom radial meshing for fuel cells performs incorrectly
6217	MPACT	Use of 'edits' card across multiple 'STATE' blocks results in some default edits not being included in HDF5 output
6337	MPACT	Summary file Assembly Exit Temperatures and Densities are incorrect for rodded assemblies
6346	MPACT	Incorrect number of axial elements for data sets in HDF5 files with homogenized grids
6347	MPACT	Steady-state calculations segfault with transient options set in MPACT block * WORKAROUND: Comment out all transient options in MPACT block
6339	MPACT	Cycle exposure incorrect if data is read in using restart_read for multiple states
6340	MPACT	Search options are not imported from a restart file
6384	MPACT	Summary file incorrectly reports cycle exposure as "core exposure"
3743	CTF	The channel splitting feature of CTF enables a user to split a large channel into several small channels in the axial direction or, vice versa, condense several small channels into one large channel. This can be done only in multi-axial-section models. It has been observed that the code may create mass in the system when using this feature for transient simulations.
3307	CTF	The grid droplet breakup model is used to divide the droplet field into large and small drops due to impacting with spacer grids in loss-of-coolant accident conditions. An uninitialized variable has been detected in this model that leads to unpredictable results. This is an experimental feature and is minimally tested, so its use is cautioned.
3788	CTF	There is a bug in the way that the calculated critical heat flux is being time-relaxed for cases that use the W-3 correlation. This bug only affects models using the W-3 correlation and only transient simulations. Steady-state cases will still arrive at the same answer as if this bug did not exist.
3321	CTF	The CTF legacy restart feature is not currently working.
3731	CTF	The droplet de-entrainment model for annular mist flows that travel into top quench fronts is not working correctly. The wetted perimeter that is used as input to this model is not calculated correctly. This model would only be encountered in accident condition simulations (loss-of-coolant accident).

3763	CTF	For cases with reversed flow at the outlet, when using the outlet pressure boundary condition, and with outlet voids greater than 0.2, it is possible that the outlet void boundary condition may become inconsistent and lead to the code crashing.
3932	CTF	Using an inlet pressure/outlet pressure boundary condition has been observed to lead to discontinuous mass flow rates in the inlet momentum cell.
3978	CTF	Form losses specified for the inlet plane of the model will not be captured in the pressure losses when using an inlet mass flow rate boundary condition.
3791	CTF	Pressure distribution is discontinuous at the inlet boundary when modeling cases with reversed flow.
3938	CTF	CTF predicted axial temperature distribution is nonlinear for cases with a uniform axial power distribution when using a nonuniform axial mesh.
4231	CTF	The fine-mesh renoding capability, used for quench front tracking on fuel rod surfaces, does not work.
4372	CTF	The code crashes when attempting to include a heated plate geometry in the model.
4442	CTF	Radiative heat transfer model input does not work.
5285	CTF	The code is very susceptible to roundoff error for long-running transients. It has been observed that some tests will fail in different builds or on different machines just due to roundoff error.
6309	CTF	Error in xml2ctf when calculating the baffle thickness due to unit error.
6359	CTF	CTF calculates velocities using the previous timestep void and density.
6081	VERAIO	Multiassembly BWR models generated using Xml2ctf will not have solution data printed correctly to the VERA HDF5 file.
5515	VERAIO	Some multiassembly BWR models containing large water rods are not being generated correctly by Xml2ctf, leading to the model crashing in CTF.
6112	VERAIO	Xml2ctf will crash when modeling a BWR assembly where the channel round radius is larger than the corner channel width.
6181	VERAIO	Xml2ctf only supports one assembly type for form loss coefficients (FLCs) in PWR models. One assembly may have different grids at different axial locations; each grid having a different FLC. However, different assemblies in the radial direction may not have different FLCs from the first unique assembly type. The user may enter assemblies with different FLCs in in VERAIn, but Xml2ctf will use the first unique assembly to define the FLCs in all assemblies.
N/A	VERAIO	Canopy might not run on some of the most recent Linux distributions. This is based on feedback from various users. VERAView
6282	VERAIO	VERAView X/Y labels switched for B&W plants
5882	VERAIO	Improved Error Messages from VERAIn
5990	VERAIO	Multiple assembly IDs can be defined in the same input
6205	VERAIO	User errors with multi-line input cards can lead to unexpected behavior or errors
N/A	VERAShift	Out of memory errors will occur if using unique pins for various quarter and full core problems *WORKAROUND: Do not use unique pins (only transfer the fission source)
5808	VERAShift	Out of memory errors and other segfault errors can occur when using a secondary source definition and performing detector response calculations
6004	VERAShift	Moderator density defaults to 1 g/cc in core, bypass, and downcomer regions if <i>modden</i> is not specified in a coupled case
5444	VERAShift	State parameters (inlet temperature and density, boron concentration, rod positions) beyond the first state are not updated in VERAShift for problems with more than one statepoint
6056	VERAShift	VERAShift execution may suspend or 'hang' during the Monte Carlo transport solve due to long histories or bad weight windows *WORKAROUND: Refine the weight window mesh by decreasing the <i>refl_mesh_size</i> and/or the <i>extend_axial_mesh_size</i> parameters
6057	VERAShift	VERAShift segfaults when a <i>restart_write</i> is present in a STATE block
6221	VERAShift	VERAShift dies ungracefully when specifying tally outside reflecting boundary for quarter core problems
6248	VERAShift	VERAShift stalls when optimizing for the bioshield using CADIS
6272	VERAShift	Pin adjoint in VERAShift output is incorrect for 2-loop B&W plants

6013	MAMBA	Axial remeshing is not supported for CRUD shuffling
N/A	BISON	The number of processors used by the libmesh build process within MOOSE is controlled by the environment variables MOOSE_JOBS and LIBMESH_JOBS. If building on less than 8 cores, it is advisable to set these environment variables equal to the number of available processors.
N/A	BISON	The standalone BISON test suite is experiencing EXODIFF failures. The root cause has been traced to an issue with test output files missing a t=0 time step point in the exodus file compared to reference "gold" files. This issue is related to the tests themselves and is not believed to be reflective of a code failure.
N/A	Dakota	Some methods that write intermediate files (e.g. LHS.err) can't be run as concurrent iterators.
N/A	Dakota	dprepro doesn't support the full range of permitted variable descriptors or string variable values. (Workaround: use variables names currently accepted: \w, which is [a-zA-Z_]).
N/A	Dakota	Support for categorical variables missing from several methods.
N/A	Dakota	Importing tabular files into Matlab no longer works straightforwardly due to presence of interface ID.
N/A	Dakota	Separate work directories not created for concurrent iterators.
N/A	Dakota	D-optimal sampling designs do not work reliably.
N/A	Dakota	Colony COBYLA methods may not return optimal solution.
N/A	Dakota	Sequential hybrid method may only propagate solutions with explicit model specification; use model_pointer as a workaround.
N/A	Dakota	Initial points are ignored by surrogates built from imported data.
N/A	Dakota	Efficient global method may not produce tabular output.
N/A	Dakota	PCE methods do not produce statistics when used with design variables; use uncertain instead.
N/A	Dakota	User interrupt may not reliably terminate. Dakota when running moga/soga methods.
N/A	Dakota	Discrete to continuous variable mapping does not work in nested studies.
N/A	Dakota	Global Reliability's input spec has invalid keywords.

REFERENCES

1. Turner, J.A., Clarno, K., Sieger, M., Bartlett, R., Collins, B., Pawlowski, R., Schmidt, R., and Summers, R., “The Virtual Environment for Reactor Applications (VERA): Design and architecture,” *Journal of Computational Physics*, 326, 544-568 (2016).
2. MPACT Development Team, “MPACT Theory Manual, Version 2.0.0”, Oak Ridge National Laboratory and University of Michigan, CASL-U-2015-0078-000 (2015).
3. I. C. Gauld, G. Radulescu, G. Ilas, B. D. Murphy, M. L. Williams, D. Wiarda, “Isotopic Depletion and Decay Methods and Analysis Capabilities in SCALE”, *Nuclear Technology* 174(2) p169–195 (2011).
4. Salko, R.K., and Avramova, M.N., “COBRA-TF Subchannel Thermal-Hydraulics Code (CTF) Theory Manual, Revision 0,” CASL Technical Report CASL-U-2015-0054-000 (2015).
5. R. Williamson et al., “Multidimensional multiphysics simulation of nuclear fuel behavior,” *Journal of Nuclear Materials*, 423: pp. 149-163 (2012).
6. Pandya, T.M., Johnson, S.R., Evans, T.M., Davidson, G. G., Hamilton, S.P., Godfrey, A.T., “Implementation, Capabilities, and Benchmarking of Shift, a Massively Parallel Monte Carlo Radiation Transport Code,” *Journal of Computational Physics*, 308, 239-272 (2015).
7. B. Collins, W. Gurecky, et al. “MAMBA v2.0 Manual”. Oak Ridge National Laboratory. CASL-U-2019-1836-000. (2019)
8. Palmtag, S., Godfrey, A., Baird, M., and Walker, E., “VERA Common Input User Manual,” CASL Technical Report CASL-U-2014-0014-004 (2019).
9. Godfrey, A., Lee, R., Lange, T., “VERAView User’s Guide”, CASL Technical Report CASL-U-2016-1058-001 (2016).
10. Godfrey, A. “VERA Core Physics Benchmark Progression Problem Specifications”, CASL Technical Report CASL-U-2012-0131-004 (2014).
11. G. Mangham, et al. “Benchmark of VERA Predictions at Steady State Conditions vs. Measurements at Westinghouse Nuclear Power Plants”. Westinghouse Electric Company. CASL-U-2019-1904-000. (2019).
12. “VERA Software Quality Assurance Plan”. Oak Ridge National Laboratory. VERA-QA-002 Rev 1. (2019)
13. Adams, B.M., Bauman, L.E., Bohnhoff, W.J., Dalbey, K.R., Ebeida, M.S., Eddy, J.P., Eldred, M.S., Hough, P.D., Hu, K.T., Jakeman, J.D., Stephens, J.A., Swiler, L.P., Vigil, D.M., and Wildey, T.M., “Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.0 User’s Manual,” Sandia Technical Report SAND2014-4633, July 2014. Updated May 2017 (Version 6.6).
14. Godfrey, A., et al., “VERAOut – VERA HDF5 Output Specification”, CASL Technical Report CASL-U-2014-0043-001 (2014).
15. Lee, R., Godfrey, A., “VERAView API and Programmer’s Guide”, CASL Technical Report CASL-U-2019-1893-000 (2019).