VERA Virtual Environment for Reactor Applications

ORNL/SPR-2022/2533

VERA 4.3 Release Notes

September 12, 2022

Mark Baird[1], Ben Collins[2], Andrew Godfrey[1], Aaron Graham[1], Brendan Kochunas[3], Ron Lee[1], Robert Lefebvre[1], Bob Salko[1], Erik Walker[1], Tara Pandya[1], and William Gurecky[1]

[1]Oak Ridge National Laboratory
[2]Veracity Nuclear
[3]University of Michigan

Approved for public release.
Distribution is unlimited.

OAK RIDGE National Laboratory

U.S. DEPARTMENT OF ENERGY

ORNL is managed by UT-Battelle LLC for the US Department of Energy.

ORNL/SPR-2022/2533

VERA 4.3 RELEASE NOTES

Mark Baird[1], Ben Collins[2], Andrew Godfrey[1], Aaron Graham[1], Brendan Kochunas[3], Ron Lee[1], Robert Lefebvre[1], Bob Salko[1], Erik Walker[1], Tara Pandya[1], and William Gurecky[1]

[1]Oak Ridge National Laboratory
[2]Veracity Nuclear
[3]University of Michigan

Date Published: September 12, 2022

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831-6283
managed by
UT-Battelle, LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

VERA 4.3 Release Notes

Revision Log

Revision	Date	Affected Pages	Revision Description
0	September 12, 2022	All	Initial VERA 4.3 release

Document pages that are:

Export Controlled:	None
IP/Proprietary/NDA Controlled:	None
Sensitive Controlled:	None
Unlimited:	All

CONTENTS

LIST OF TABLES

ABBREVIATIONS

1G	1-group
ATF	accident–tolerant fuel
BWR	boiling-water reactor
CASL	the Consortium for Advanced Simulation of Light-water Reactors
CFD	computational fluid dynamics
CHF	critical heat flux
CILC	CRUD-induced localized corrosion
CIPS	CRUD-induced power shift
CMFD	coarse mesh finite difference
COBRA-TF	COolant Boiling in Rod Arrays – Two Fluid
CRUD	Chalk River unidentified deposit
GUI	graphical user interface
HDF5	Hierarchical Data Format 5
IFBA	integral fuel burnable absorber
INL	Idaho National Laboratory
JFNK	Jacobian–free Newton–Krylov
LWR	light-water reactor
MC	Monte Carlo
MG	multi-group
MLCMFD	multilevel CMFD
MOC	method of characteristics
MPI	Message Passing Interface
NEM	nodal expansion method
NQA-1	Nuclear Quality Assurance 1
P_N	spherical harmonics
PCI	pellet–clad interaction
PWR	prssurized-water reactor
QA	quality assurance
RIA	reactivity insertion accident
RSICC	Radiation Safety Information Computational Center
SENM	source expansion nodal method
SMR	small modular reactor
SP_N	simplified P_N
SQL	software quality level
SQL1	software quality level 1
SQL2	software quality level 2
SQL4	software quality level 4
STH	simplified TH
T/H	thermal hydraulics
TCP_0	transport–corrected P_0
TIP	transverse in-core probe
TPL	third party libraries
UQ	uncertainty quantification
V&V	validation and verification

VERA	Virtual Environment for Reactor Applications
VERAIn	VERA Common Input
VTK	Visualization Toolkit
VUG	VERA Users Group
XS	cross section

1. INTRODUCTION

The Virtual Environment for Reactor Applications (VERA) components contained in this distribution include selected computational tools and supporting infrastructure that solve neutronics, thermal hydraulics (T/H), fuel performance, ex-core radiation transport, and Chalk River unidentified deposit (CRUD)/chemistry problems for commercial light-water reactors (LWRs). In many cases, these tools can be executed standalone 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 LWR 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, 3]. For 3D problems, MPACT uses the 2D/1D method, which uses 2D MOC in each radial plane and 1D spherical harmonics (P_N) in the axial direction. MPACT includes integrated cross section (XS) capabilities that provide problem-specific 51 energy group XSs generated using the subgroup self-shielding methodology. MPACT also includes ORIGEN's isotopic depletion and decay capability [4] for the simulation of all reactor operating regimes as a function of time or burnup. MPACT also handles the simulation of other aspects of reactor operation, such as control rod movement, fuel shuffling, refueling outage decay, and in-core instrumentation response. The code can solve both 2D and 3D problems on parallel processors to significantly reduce overall run time.

A T/H capability is provided with CTF, an updated version of COolant Boiling in Rod Arrays – Two Fluid (COBRA-TF) [5], that allows T/H subchannel analyses for single and multiple fuel assemblies using the simplified VERA Common Input (VERAIn). This distribution also includes coupled neutronics, T/H, and depletion capabilities to allow calculations using MPACT coupled with CTF and ORIGEN, and it includes a 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 [6] calculates—on a 1D, 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 enable 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).

BISON is not included in VERA 4.3 distributions through the Radiation Safety Information Computational Center (RSICC). To obtain BISON, please contact `agradmin@inl.gov`.

VERA 4.3 is distributed with the Shift Monte Carlo (MC) radiation transport code [7], which can be run standalone using a VERAIn file with the *vera_shift* executable and directly coupled to MPACT/CTF using the *vera_to_shift* executable. VERAShift uses the SCALE ENDF/B-VII.1 continuous-energy (CE) data library and has been used to model a broad set of problems for the the Consortium for Advanced Simulation of Light-water Reactors (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 validation and verification (V&V) report shown in Section 5. VERAShift has been successfully demonstrated for ex-core detector responses and vessel fluence applications.

VERA 4.3 also includes the MAMBA [8] 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 subcooled boiling duty from CTF. MAMBA is also capable of system mass balance of the CRUD sources as well as CRUD removal and shuffling during refueling outages. A computational fluid dynamics (CFD)–informed subchannel methodology implemented in CTF provides the capability of producing very detailed local CRUD deposits with consideration for complex fluid behavior downstream from spacer grids with mixing vanes. MAMBA provides a CRUD-induced power shift (CIPS) and CRUD-induced localized corrosion (CILC) analysis capability, but it is still under development and requires more testing, calibration, and validation prior to end user applications.

Input processing capabilities include the VERAIn processor, which provides a simple and user-friendly input interface for all VERA components [9]. VERAIn converts the ASCII common input file to an intermediate XML format and serves as a common geometric database for each of the physics components in VERA. VERA component codes read either the VERA XML format directly or provide a preprocessor that can convert the XML into native input.

VERAView [10] is a graphical user interface (GUI) for the visualization and engineering analyses of output data from VERA. The Python-based software is easy to install and intuitive to use, and it provides instantaneous 2D and 3D images, 1D plots, and alphanumeric data from VERA multiphysics simulations. VERAView is open-source and can be obtained from GitHub at `https://github.com/CASL/VERAview` or by contacting `vera-support@ornl.gov`.

VERA 4.3 has been extensively tested against a broad series of small-scale, automated tests as part of its continuous-integration and build development environment. Full-scale commercial power plant models have also been tested, but the breadth of these tests is limited. Generally, the following commercial power plants have been successfully simulated or analyzed to some degree with VERA 4.3:

- Watts Bar Unit 1 Cycles 1–3
- Five modern fuel cycles of a publicly available B&W plant
- APR-1400 Cycle 1

VERA 4.3 has also been successfully used to generate solutions to the CASL core physics benchmark progression problems [11], a publicly available set of test problems based on Watts Bar Unit 1 that includes plant measurements and high-fidelity reference solutions for software testing and validation. The VERA 4.3 results for these problems can be provided upon request.

The following components of VERA 4.3 are not recommended for use by end users due to the current level of maturity or lack of sufficient testing. These items are not supported software quality level 1 (SQL1) capabilities (see Section 2).

- MAMBA: The MAMBA capabilities in VERA 4.3 are still under active development and should not be used or tested by end users. To access a more capable and tested version of MAMBA, please contact `vera-support@ornl.gov`.
- VERAOneWay scripts and processors: The one-way coupling software that supports the generation of BISON or FAST inputs from VERA output data for all fuel rods over multiple fuel cycles is not included in the VERA quality assurance (QA) program. This software is still under active development and is not recommended for use in VERA 4.3. More recent software is available for testing, if desired. Please contact `vera-support@ornl.gov`.
- VERAShift: A software issue in the development of VERA 4.3 resulted in a substantial memory increase for VERAShift applications. The new memory usage is approximately double that of VERA 4.1. Please contact `vera-support@ornl.gov` for assistance in executing VERAShift applications in VERA 4.3.

Additionally, VERA 4.3 contains experimental support for boiling-water reactor (BWR) calculations. In general, neutronics calculations are expected to work consistently but may have poor computational performance with respect to runtime or memory. Coupled calculations with CTF are not recommended due to excessive runtime and issues with numerical stability. However, the simplified TH (STH) capability should work in many cases to provide lower fidelity T/H feedback to the neutronics calculation. Many common BWR geometry components are implemented such as control blades, thick-thin channel boxes, and large water rods.

2. SOFTWARE QUALITY ASSURANCE

The VERA 4.3 software product suite was developed, validated, and tested under a Nuclear Quality Assurance 1 (NQA-1) QA program established in 2019 [12]. CASL initiated this QA program based on nuclear industry feedback identifying the NQA-1 standard, which is used throughout the US nuclear industry, as a top priority to enable the adoption and commercialization of VERA.

The VERA software components included in this distribution are still under active development and are subject to change. However, all software life cycle work activities are performed, managed, and controlled under the NQA-1 compliance 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 (TPL) and utilities that are used to support the SQL1 codes but do not directly implement the theoretical model are designated as software quality level 2 (SQL2). Acceptance for their use is documented in core product software management plans, baseline software documentation, and testing protocols.
- **SQL4**: Software in the R&D stage are included under the NQA-1 program as software quality level 4 (SQL4) to drive toward mature processes, controls, and documentation consistent with SQL1. MAMBA, VERAShift, and VERAOneWay are included in the distribution of VERA 4.3 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. The BISON version used in VERA 4.3 should be considered for test and evaluation purposes only (SQL4). To obtain a more recent version of BISON, please contact agradmin@inl.gov.

The SQL1 codes CTF, MPACT, and VERAIO contain mature and research capabilities. The mature capabilities are rigorously implemented and tested before software release. The research capabilities are still under R&D and have not been thoroughly tested, demonstrated, or validated. These R&D capabilities are not supported features of VERA 4.3 but may be more mature in future releases. Examples include the following.

- Examples of supported features: prssurized-water reactor (PWR) steady-state core follow, depletion, decay, fuel shuffling, xenon transients, PWR standalone T/H analyses, and single-phase flow.
- Examples of unsupported R&D features: BWRs, fully coupled transients such as reactivity insertion accident (RIA), ex-core neutron transport calculations, CRUD analyses (CIPS and CILC), CFD-informed subchannel applications, and pellet–clad interaction (PCI) risk assessments.

During release testing of VERA 4.3, the code development teams identified and addressed several defects and issues. Some of these were corrected in VERA 4.3, and some will be addressed in later VERA versions. Section 9 contains a comprehensive list of known defects, issues, and workarounds in this release. Please report any additional new defects identified to vera-support@ornl.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 of fidelity consistent with CASL analyses, a minimum of approximately 500 compute cores are needed with approximately 4 GB memory available per core. 1,000 cores are recommended for these problems. When running VERA with VERAShift ex-core calculations, another 100–400 cores are recommended, and the memory requirement per process is approximately doubled.

Detailed system software and TPL 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. Documentation is also delivered with the distribution.

5. DOCUMENTATION

The documentation listed in Table 1 is available with the VERA 4.3 release distribution. Other CASL and VERA technical reports and publications are available upon request from vera-support@ornl.gov.

Table 1. Caption

Document ID	Document Title
ORNL/TM-2022/2523	VERA 4.3 Installation Guide
ORNL/SPR-2022/2509	VERAIn User's Manual
ORNL/SPR-2022/2502	VERAView User's Manual
ORNL/SPR-2022/2480	VERAIn Programmer's Manual
ORNL/SPR-2022/2488	VERAView Programmer's Manual
ORNL/SPR-2022/2494	CTF Theory Manual
ORNL/SPR-2022/2495	CTF User's Manual
ORNL/SPR-2022/2500	CTF Verification and Validation Manual Manual
ORNL/SPR-2021/2330	MPACT Theory Manual
ORNL/SPR-2021/2331	MPACT User's Manual
ORNL/SPR-2021/2332	MPACT Verification and Validation Manual Manual
ORNL/SPR-2021/2353	VERAShift Theory Manual
ORNL/SPR-2021/2357	VERAShift User's Manual
ORNL/SPR-2021/2355	VeraShift Verification and Validation Manual
CASL-U-2016-1186-000	Shift Verfication and Validation Manual
ORNL/SPR-2022/2524	MAMBA Theory 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-2017-1445-000 / 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)

6. SUPPORT

Questions, issues, bugs, and suggestions should be reported to `vera-support@ornl.gov`. Some documentation and examples are also available at `https://vera.ornl.gov`. Software builds and examples are available at */projects/vera-user/vera* on INL's Sawtooth high-performance computer and for VERA Users Group (VUG) members at */projects/vera-users-grp*.

7. PHYSICS COMPONENTS INCLUDED IN VERA 4.3

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 component code version history.

Component	VERA 4.0 Version	VERA 4.1 Version	VERA 4.2 Version	VERA 4.3 Version
MPACT	b5e2cbe (Mar 20, 2019)	869ab07 (Oct 17, 2019)	9fa11cb (Dec 1, 2020)	18f6e2c (Jul 20, 2022)
CTF	30aab7c (Mar 25, 2019)	97514a5 (Nov 14, 2019)	ee32ff4 (Jan 29, 2021)	69e3e14 (Jul 11, 2022)
VERAIO	N/A	ac2e962 (Oct 18, 2019)	555ad61 (Dec 2, 2020)	2ceb8d2 (Jul 13, 2022)
BISON	be31f34 (Feb 26, 2019)	68913d5 (May 29, 2019)	f721711 (Jun 11, 2020)	fcaf877 (Jun 1, 2021)
VeraShift	911aa80 (Feb 14, 2018)	cc034ad (Mar 19, 2019)	ac561d8 (June 6, 2019)	e901353 (Dec 21, 2020)
MAMBA	d971461 (Mar 25, 2019)	6bd9ab2 (May 22, 2019)	c9e37e4 (Oct 17, 2020)	ad795e0 (Sep 16, 2021)
Dakota	V6.6	V6.6	V6.6	V6.6

Complete version information for all VERA source code repositories associated with this release is shown below. For VERAIO, the version of VERARun supporting the 4.3 release is Version 1.11. The SQL1 version of VERAView is Version 2.4.3. Any subsequent versions should be considered test or R&D versions until documented in a subsequent release. For BISON, the compatible commit SHA is fcaf877748c406e86cee 772451c3f6059c0bd453.

```
*** Base Git Repo: VERA
a896f8f [Thu Jul 7 13:20:22 2022 -0400] <bairdml@ornl.gov>
Adjusting checkout and submodule command fort production pipelines
*** Git Repo: TriBITS
9d0198a [Tue Apr 6 18:41:53 2021 +0000] <collinsbs@ornl.gov>
Merge branch 'timeout_issues' into 'master'
*** Git Repo: Trilinos
3b9c7e3 [Thu Jan 30 00:18:02 2020 -0500] <collinsbs@ornl.gov>
Merge pull request #1 from bartlettroscoe/casl-phi-6249-upgrade-tribits-trilinos
*** Git Repo: TeuchosWrappersExt
781aae8 [Fri May 10 09:51:36 2019 -0600] <rabartl@sandia.gov>
Add missing std include (PHI-5294)
*** Git Repo: Futility
2c86eec [Wed Jun 15 20:55:43 2022 +0000] <salkork@ornl.gov>
Fix to the requirements generator script
*** Git Repo: MAMBA
```

```
ad795e0 [Thu Sep 16 04:00:43 2021 -0400] <collinsbs@ornl.gov>
Merge branch 'master' of code-int.ornl.gov:mamba/MAMBA
*** Git Repo: VERAIO
2ceb8d2 [Wed Jul 13 11:26:28 2022 -0400] <ew4@ornl.gov>
Add tech reviewer changes to VERAIn User's Manual
*** Git Repo: DataTransferKit
19526ab [Fri Aug 18 09:14:16 2017 -0400] <dalg24@gmail.com>
Merge pull request #297 from naughtont3/tjn-fortran-off
*** Git Repo: COBRA-TF
69e3e14 [Mon Jul 11 12:32:30 2022 +0000] <salkork@ornl.gov>
Updates to the CTF documentation for the 4.3 release
*** Git Repo: VERAData
eb21c20 [Thu Jun 3 13:29:31 2021 +0000] <grahamam@ornl.gov>
Merge branch '4251_MPACT_v5.1m0_60n19g' into 'master'
*** Git Repo: VERAData/CEData
8252504 [Wed Jul 8 20:44:25 2020 -0400] <pandyatm@ornl.gov>
Merge branch 'update-hydrogen-pole-data' into 'master'
*** Git Repo: VERAOneWay
6af5147 [Tue Sep 14 13:46:59 2021 +0000] <collinsbs@ornl.gov>
Merge branch 'bison_post_states' into 'master'
*** Git Repo: SCALE
1a8cd5c [Tue Aug 17 12:12:19 2021 +0000] <pandyatm@ornl.gov>
Merge branch 'sync-scale' into 'master'
*** Git Repo: XSTools
08adb77 [Mon May 24 20:33:17 2021 +0000] <collinsbs@ornl.gov>
Merge branch '4874_XSTools_wrong_backXS_5x' into 'master'
*** Git Repo: MPACT
18f6e2c [Wed Jul 20 15:32:04 2022 -0400] <grahamam@ornl.gov>
Fix possible out of bounds error
*** Git Repo: VeraShift
767cddc [Fri Aug 27 10:44:46 2021 -0400] <bairdml@ornl.gov>
Updating trigger for DownStream testing.
*** Git Repo: DakotaExt
afd354f [Tue Aug 1 19:36:55 2017 -0400] <rhoope@sandia.gov>
Fix tarball installation
*** Git Repo: DakotaExt/Dakota
7415238 [Thu Jun 15 12:40:49 2017 -0400] <rhoope@sandia.gov>
Triage static builds of Dakota 6.6
*** Git Repo: VUQDemos
b2d0175 [Tue Oct 8 14:13:19 2019 -0400] <salkork@ornl.gov>
Merge branch '1-correcting-hardcoded-python-paths-in-vuq-core' into 'master'
*** Git Repo: VERAView
28e0f8a [Thu Mar 31 16:53:16 2016 -0400] <leerw@ornl.gov>
Sync from code.ornl.gov for version 1.0
```

7.1 MPACT

MPACT is an advanced pin-resolved, whole-core, multigroup deterministic neutron transport capability based on the 2D/1D synthesis method, on the frame of a 3D coarse mesh finite difference (CMFD) method, for which radial and axial correction factors are obtained from 2D MOC and 1D P_N, respectively [2]. The transport is performed using 51 energy group XSs, based on the subgroup method of on-the-fly resonance self-shielding [13]. The discretization of the core is typically three radial and eight azimuthal flat source regions per fuel pellet at each of approximately 60 axial planes, explicitly treating such features as spacer grids, fuel and absorber plena, and end plugs. MPACT performs the same neutron transport calculations in the upper, lower, and radial reflector regions of the core, explicitly modeling the baffle, core barrel, neutron pads, nozzles, and core plates. It requires no *a priori* approximations of the core boundary conditions as is needed for nodal diffusion methods. MPACT also controls the functional application features of the VERA core simulator, such as critical boron search, equilibrium xenon calculations, predictor–corrector depletion, in-core detector response calculations, reading and writing restart files, and performing fuel shuffling, decay, and discharge.

New features implemented since the release of VERA 4.2 include the following:

- various bug fixes
- capability to write pin-wise decay heat to Hierarchical Data Format 5 (HDF5)
- experimental capability to perform jump-in calculations for BWRs using ORIGEN
- BWR transverse in-core probe (TIP) detector inputs and processing
- additions to summary file for BWR-related quantities such as void, thermal limits, etc.
- cell-based XS shielding capability for PWRs to improve performance
- source expansion nodal method (SENM) as an additional axial solver option
- new STH models for BWRs
- native (Futility-based) solvers for CMFD
- geometry improvements to support BWR modeling
  - BWR control blades
  - thick-thin channel boxes
  - default radial reflector options for BWRs
  - support for quarter and full symmetry
  - support for part-length fuel rods
  - support for large water rods
  - memory improvements
- mixed-precision storage
- improvements to linear source MOC stability and meshing options
- adaptive v-cycle capability to multilevel CMFD (MLCMFD)
- STH modified to use IAPWS water properties to increase consistency with CTF
- Improvements to hexagonal modeling
  - Support multiple pins
  - CMFD
  - 2D/1D
  - parallel calculations
- gamma transport
- gamma detector response
- multiscale T/H coupling using STH as an accelerator for CTF
- new *[TRANSIENT]* input block to improve transient modeling
- checkpoint file for transient calculations

New features implemented since the release of VERA 4.1 include the following:

- various bug fixes
- the approximate gamma-smearing energy deposition model

- additions to nodal XS edits for transient data calculation

New features implemented since the release of VERA 4.0 include the following:

- various bug fixes

New features implemented since the release of VERA 3.9 include the following:

- various bug fixes
- updated documentation
- in-line thermal expansion processing (the 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 MOC solver
- addition of ^{15}N to XS library
- nodal XS 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 Message Passing Interface (MPI)
- parallel angular decomposition with MPI
- user-defined macroscopic XSs
- 51-group macroscopic XS library data
- subgroup resonance self-shielding
- transport-corrected P_0 scattering treatment
- export of mesh to legacy Visualization Toolkit (VTK) and VTU file for visualization
- 2D MOC transport kernel
- CMFD acceleration
- 1D nodal kernels based on nodal expansion method (NEM) and SENM for diffusion and simplified P_N (SP_N)
- 2D/1D full-core solution
- multistate calculation capability
- transient calculation capability
- depletion and decay
- critical boron search
- equilibrium xenon calculation
- direct coupling with COBRA-TF
- STH
- general PWR geometry modeling
  - integral fuel burnable absorber (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
- jump-in cycle capability
- radial thermal expansion
- efficient graph-based spatial decomposition
- generation of text summary file
- in-line thermal expansion
- nodal XS edits

The following features are considered to be experimental:

- transient checkpoint file
- gamma transport and detector response
- hexagonal geometry
- MLCMFD
- mixed-precision storage
- native CMFD solvers
- cell-based XS shielding capability
- isotopic jump-in capability
- separate ^{10} depletion of soluble boron
- space-dependent Wielandt shift or dynamic Wielandt shifts
- modeling of BWRs
- simplified CRUD modeling
- post-corrector depletion
- 3D MOC solvers
- processing of AMPX working XS libraries
- processing of ISOTXS XS libraries
- secondary source calculation
- axial thermal expansion
- multilevel 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
- linear source MOC
- support for accident–tolerant fuel (ATF) fuel forms
- explicit delayed energy deposition for transients
- intra-pin edits for VERAView
- coupling of intra-pin power, temperature, and burnup with CTF
- secondary source calculations

7.2 CTF

CTF, an updated version of the COBRA-TF code, is a subchannel T/H code that uses a two-fluid, three-field (i.e., fluid film, fluid drops, and vapor) modeling approach for single- and two-phase steady-state and transient condition [5]. CTF includes a wide range of T/H models important for 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, two-phase pressure drop, spacer grid effects, and void drift. Because of its 3D capabilities and extensive array of reactor T/H modeling capabilities, CTF has been used extensively in the modeling of LWR in-core, rod bundle transient analyses.

Significant changes implemented since the release of VERA 4.2 include the following:

- added a restart capability, which allows the user to dump restart data to a restart file at select intervals in the transient or steady-state solution and then start the solution by reading from the restart file
- exposed the restart capability to the CTF Coupling Interface
- allowed Xml2ctf to accept BWR channel box radii that are larger than the corner channel width
- modified Xml2ctf so that it recognizes the keyword "cool" as moderator material in the VERAIn file
- expanded the CTF Coupling Interface by providing procedures for transferring data between different mesh refinement levels, which is used when coupling to a CTF nodal resolution model
- added verbosity level for CTF outputs, allowing the user to make warning messages more or less detailed
- added a new annular flow transition model for predicting the transition from bubbly to annular mist flow regime
- created a system coupling interface and used it to perform a simple coupling between CTF and the system code TRACE
- added support for modeling BWR cores in Xml2ctf that have assemblies with different fuel lattice sizes
- added support to Xml2ctf for modeling of water rods with dimensions that change relative to axial location
- added support for using up to 4 processors per assembly in BWR models, which allows for higher domain decompositions and faster BWR simulations when performed in parallel
- modified Xml2ctf to consider bypass flow and relative STATE flow in the VERAIn file when setting flow in the CTF model
- updated the default compiler for CTF to GCC 8.3.0
- added support for modeling of the lower tie plate losses in BWR models in CTF and expanded Xml2ctf to read this information from the VERAIn file
- applied the ROTHCON grid heat transfer model to the conduction solve solution instead of just the CRUD growth calculation
- added ability to set inlet subcooling as the CTF initial condition and boundary condition
- implemented new drift-flux based interfacial drag model for the annular mist flow regime
- added a simplified two-phase, steady-state, drift-flux–based subchannel solver called Alternative Nonlinear Two-phase Solver (ANTS) as a solver option in CTF
- added a feature to Xml2ctf for generation of nodal resolution CTF models (4 subchannels per assembly)
- added a feature to disable the solution of the noncondensable gas field in CTF
- added an option for Xml2ctf to place spacer grids at the nearest mesh cell face rather than forcing the user to mesh the spacer grids explicitly
- added an option for writing more detailed T/H data to the VERA HDF5 file
- Move the location of the grid heat transfer enhancement from the bottom of the spacer grid to the top of the spacer grid when building a model with Xml2ctf
- Fix a bug in the calculation of the fuel rod gap in Xml2ctf, which was double-counting the impact of thermal expansion
- performed various bug fixes

Significant changes implemented since the release of VERA 4.1 include the following:

- added a preliminary bypass modeling capability for core-scale BWR models
- made performance and robustness improvements to pressure balance iteration loop used for core-scale BWR models
- added a preliminary capability for modeling axially varying water rod geometry to the xml2ctf preprocessor
- added inlet orifice map feature for BWR models
- set energy storage printout to zero if it is not being checked to prevent confusion
- added alternative Chisholm and Lockhart–Martinelli two-phase pressure drop models
- added ability to specify which datasets CTF prints to the HDF5 file through the *CTF_Coupling_Interface*
- switched to solving one pressure matrix per assembly for core-scale BWR models, which allows for using the direct solver without an impact on problem run time
- added basic support for using the Intel 19 compiler for CTF builds
- refactored flow regime map logic to allow for implementing new maps and implemented the Wallis flow regime map
- implemented an outer-iteration loop, which iterates over the mass and energy equation solution to ensure that they are sufficiently converged before advancing to the next timestep
- added preliminary drift-flux model for predicting interfacial drag in bubbly flow regimes
- added support for refining the radial mesh in the fuel rod clad
- implemented a set of common ATF clad and pellet material properties and refactored the material properties section to allow for more easily implementing new properties in the future
- expanded CTF validation matrix to include WALT, Thom, and Rohsenow boiling tests and Bartolomei void measurement tests
- added a validation matrix driver, which allows users to run the entire CTF validation matrix and perform results postprocessing in an automated fashion
- added new subcooled boiling model that uses an onset of nucleate boiling criteria with the Gorenflo heat transfer model and the Saha–Zuber critical enthalpy model
- added option to output solver residuals during solution
- added support for modeling quarter-symmetry BWR models
- added stopping criteria for species transport non-condensable gas model, allowing users to model steady-state solutions of species transport of noncondensable gas
- added new set of *CTF_Coupling_Interface* procedures that do not expose the cell splitting behavior for fuel rods in CTF, thus simplifying the interface
- added support for modeling of part-length rods in the xml2ctf preprocessor
- modified the external fuel solver coupling interface to support coupling to the FAST fuel performance code

Stable features include the following:

- solid modeling capabilities
  - radial conduction
  - nuclear fuel rod models (e.g., pellet, gap, and clad regions and UO_2 and Zircaloy material properties)
  - dynamic gap conductance model (e.g., pellet relocation and PCI)
- fluid modeling capabilities
  - solid-to-fluid heat transfer
    - single-phase convection
    - subcooled/saturated boiling
  - critical heat flux (i.e., 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
- modeling of axial variation of geometry in subchannels

The following features are experimental:

- solid modeling capabilities
  - fuel rod axial/azimuthal conduction
  - axial mesh refinement (quench front tracking)
  - Zircaloy–water thermal reaction
- Fluid modeling capabilities
  - noncondensable gas effects
  - post-critical heat flux (CHF) heat transfer models (are encountered in validation tests, but no validation of models done)
  - grid-directed crossflow modeling
  - boron-tracking model with consideration of boron precipitation

7.3 BISON

BISON is not included in VERA 4.3 distributions through RSICC. To obtain BISON, please contact agradmin@inl.gov.

VERA includes the capability to predict fuel rod performance by using 2D axisymmetric or 3D coupled multiphysics, and it represents a significant advancement for the modeling and analysis capabilities in LWR fuel rod behavior [6]. The capability is being constructed within the MOOSE/BISON computational framework from INL, which supports the following:

- 1.5D-RZ, 2D, and 3D thermomechanics, including elasticity, plasticity with strain hardening, creep, large strains, large displacements, and smeared plus explicit cracking
- unsteady (i.e., transient) conduction heat transfer with time- and spatial (i.e., 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)
- the use of high-performance computing platforms to achieve the parallel performance and scalability required to perform coupled multiphysics 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 the following:

- 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

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

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 could be possible; for example, chromia-doped pellets could be simulated with user-supplied models for several of the pellet performance characteristics, or steel-based clad could be simulated with similar user-supplied models. Materials such as silicon carbides that do not fit the system paradigm can be simulated but are unlikely 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. This release adds no new capability, nor does it address any significant issues from VERA 4.2. As before, VERAShift can be run standalone in eigenvalue mode using the VERAIn 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.

There are no significant changes implemented since the release of VERA 4.2.

Stable features of VERAShift in VERA 4.2 and VERA 4.3 include the following:

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

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

- transfer of temperatures and densities in fuel, clad, and coolant between VERA and Shift

- FW-CADIS mode
- domain-decomposed Shift calculation

Known limitations include the following:

- Memory usage when using unique pins is very high.
- Memory usage for full-core calculations is high.
- Memory usage when transferring fission source, isotopics, temperatures, and densities from VERA is very high, which requires unique pins.
- When running in CADIS or FW-CADIS modes, 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 [8] simulates the 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. The precipitation of boron compounds, such as lithium tetraborate ($Li_2B_4O_7$), can lead to CIPS in the nuclear fuel. Additionally, the CRUD itself can lead to CILC due to reduced thermal transport and thus increased temperatures, which can cause mechanical failure of the fuel.

MAMBA simulates the buildup of CRUD and the precipitation of boron-rich compounds within the porous CRUD layer. Because the formation of CRUD is a fundamentally multiphysics problem, MAMBA is coupled to neutronic and T/H solvers present in VERA to predict CIPS. MAMBA requires T/H conditions as input: namely, the cladding surface heat flux, turbulent kinetic energy, and the coolant temperature. Within VERA, CTF provides T/H conditions. 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 4.3, but all features are considered experimental at this time.

7.6 DAKOTA

The Dakota 6.6 package [14] manages and analyzes ensembles of simulations to provide a 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 that enable design exploration, optimization, model calibration, risk analysis, and quantification of margins and uncertainty with such models. It directly supports V&V activities. Dakota algorithms enrich complex science and engineering models, enabling analysts to answer the following crucial questions about sensitivity, uncertainty, optimization, and calibration.

- 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 the 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?

A CASL technical report that provides user guidelines and best practices for CASL validation/uncertainty quantification (UQ) 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 mature and robust:

- parameter studies: list, vector, centered, multidimensional
- UQ: 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 (e.g., genetic algorithms, direct), local least squares, surrogate-based local methods
- surrogate models: polynomials, Gaussian/kriging process, 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
- UQ: 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 multifidelity
- 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
- UQ: D-optimal sampling, multilevel and multifidelity 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 the capabilities of which 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 VERAIN

The VERAIn [9] provides a simple and user-friendly input interface for the VERA components. VERAIn is a PERL script that 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 that can convert the XML into native input (such as CTF and BISON). When processing the ASCII common input file, VERAIn performs basic error checking to ensure that common input errors are caught before running a simulation. Additionally, VERAIn 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 it provides instantaneous 2D and 3D images, 1D plots, and alphanumeric data from VERA multiphysics simulations. It reads files in HDF5 format that meet the VERAOut specification [15]. A user manual is included that provides 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 packages listed in Table 3.

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 in a prebuilt Python environment based on Miniconda (`https://docs.conda.io/en/latest/miniconda.html`) and includes 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 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 (assumes same for all data except detectors)
- 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
- NT/HETA: 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* begins at 0001 and increments thereafter. Dataset types that are recognized are summarized in Tables 4–7.

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	(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

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

Derived datasets representing the average, root mean square, standard deviation, and sum across data axes are shown in Table 5. VERAView can calculate these datasets if they are not already in the VERAOut file.

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

VERAView provides several "basic capability" and "extended capability," or experimental, widgets. The former are covered in the VERAView software test plan requirement and test report, whereas experimental widgets are lightly tested. Additionally, VERAView is designed to be extensible, supporting the addition of custom widgets. Refer to the VERAView Programmer's Manual [16].

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 and can be obtained by contacting `vera-support@ornl.gov`, or at */projects/vera user/vera/veraview/veraview3* on INL's Sawtooth.

Table 6. VERAView basic capability widgets.

Type	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.

Type	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

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 so 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 a Python package source distribution that 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 script 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 VERA input file—for example:

```
verarun 5a_2d.inp
```

VERARun is available on INL's Sawtooth by sourcing the file */projects/vera-user/vera/setenv.sh*. More information is also available in */projects/vera-user/vera/README*.

For more information on VERARun, please contact *vera-support@ornl.gov*.

9. CAVEATS AND KNOWN ISSUES

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

Issues identified during development and testing of this release package are listed in Table 8. Some of these issues are software errors (reported as part of the software problem reporting process), and some are minor issues or nuances that were encountered in the past and have not been resolved. To determine whether a particular issue affects a particular VERA use case, the user should consult the detailed software problem report (Issue #) in Table 8. Software problem reports for defects found in VERA are posted online for VUG members at `https://vera.ornl.gov/vera-members-only`. The user may also contact `vera-support@ornl.gov` regarding known or new issues with VERA use.

Table 8. Known issues and defects in VERA 4.3

Issue #	Component	Issue
3262	MPACT	For some MPI distributions, the communication routines for the multi-group (MG) MOC kernels experience an error in MPI. **WORKAROUND**: Use the *shield_moc_kernel 1g* or *moc_kernel 1g* option.
1284	MPACT	On some platforms, depletion cases fail when run with spatial and angular decomposition. **WORKAROUND**: Use threading instead of angle decomposition.
1301	MPACT	Using transport–corrected P_0 (TCP_0) 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 *moc_kernel mg* 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 flat 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.
3262	MPACT	For some MPI distributions, the communication routines for the 1-group (1G) MOC kernels experience a memory leak in MPI. **WORKAROUND**: Use *moc_kernel mg* and *moc_mg_data_passing true*.
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 CONTROL block to ensure that rod materials are not depleted. If the materials are placed elsewhere, then they might be flagged as depletable.
3995	MPACT	Assemblies with stainless-steel rods must have the same thermal expansion temperatures with performing core shuffles.
4082, 4637	MPACT	Extended coupling mesh T/H passed back from CTF not placed in MPACT properly.

Issue #	Component	Issue
3863	MPACT	Axial mesh boundaries must align with spacer grid boundaries in the active fuel region.
3865	MPACT	Smaller timesteps are required for depletion of fuel bearing the Gadolinia neutron absorber.
3795	MPACT	Subplane calculations yield inconsistent results when using graph partitioning.
3897	MPACT	Extended coupling mesh does not work when smearing axial reflector regions together.
4206	MPACT	If the number of depletable regions changes while using higher-order depletion, then MPACT will segfault.
4294	MPACT	Code segfaults instead of throwing an error if the user inputs mismatched pins on the x and y symmetry lines while using rotational symmetry.
4315	MPACT	Custom radial meshing for fuel cells performs incorrectly.
4593	MPACT	In very rare cases, the same calculation can take a different number of iterations to converge. The reason for this is unknown. When this occurs, the solution is the same, regardless of the number of iterations taken.
4693	MPACT	VTU edits might place data in the incorrect regions if the *pin_cell_mod_mesh* is used. The same issue can also occur if the *mesh* option is used to specify different numbers of azimuthal divisions for different radial regions. **WORKAROUND**: Do not use these options when visualizations are needed.
4762	MPACT VeraShift	Inconsistent Angular Input in VERAIn for MPACT and VeraShift
4846	MPACT	Heavy reflectors for small modular reactor (SMR) models sometimes have patches of water in the MPACT model that should not be there. **WORKAROUND**: Most issues are resolved by ensuring that the *baffle gap + baffle thickness = assembly gap + pin pitch*.
4877	MPACT	Restart cases sometimes crash when restarting in full symmetry from a quarter core symmetric calculation. **WORKAROUND**: Generate the restart file in full symmetry or run the restarted calculation in quarter symmetry.
4923	MPACT	Error in core-averaged exposures when modifying the radial mesh during a restart
4959	MPACT	Incorrect core exposure in summary file for high-order depletion. **WORKAROUND**: Do not use high-order depletion (experimental).
4960	MPACT	Isotopic jump-in might calculate incorrect exposure.
4967	MPACT	MPACT crashes with too many decimal places on *axial_edit_bounds*. **WORKAROUND**: Round the values of *axial_edit_bounds* to four decimal places or fewer and ensure exact consistency with fuel and grid spacer elevations.
4969	MPACT	MPACT crashes while generating output data for models with staggered fuel stacks.
4976	MPACT	*edit intrapin_isotopes_all* does not write to HDF5 file. **WORKAROUND**: List the desired isotopes instead.
4992	MPACT	Errors occur for quarter-core models when using the User CRUD capability.
5001	MPACT	MPACT thermal expansion overwrites certain fuel types if two fuels are identical, except for the *gad_mat* and *gad_frac* parameters.

Issue #	Component	Issue
5016	MPACT	Metastable isotopes are wrong for the *pin_isotopes_all* option.
5017	MPACT	O-16 bound to UO_2 (8001) is not written by *pin_isotopes_all*.
5018	MPACT	Duplicate isotope data are generated for *pin_isotopes_all*.
5019	MPACT	*pin_isotopes_all* produces data for coolant isotopes that are all 0.0 instead of generating data only for fuel rods.
5028	MPACT VeraShift	MPACT and Shift cannot process consistent isotopic input for VERAShift.
5051	MPACT	Memory defect in lumped parameter MOC XS shielding calculation can sometimes cause issues. **WORKAROUND**: Retry the calculation and it will often succeed. **WORKAROUND**: Use the 1G shielding kernel instead of the MG kernel.
5052	MPACT	Potential index-out-of-bounds error in nodal XS edits on some systems.
5060	MPACT	Incorrect calculation of velocity for nodal XS edits.
5062	MPACT	Various issues with nodal XS edits: *ASSEMBLY_XS/Heavy Metal Mass* dataset is 1/4 what it should be; incorrect unit attributes in HDF5 file for *KXSF*, *CHI*, *fluxnorm*, *ADF*, *MODFRAC*, *COOLFRAC*, and all flux, adjoint flux, current, and transverse leakage related datasets; index-out-of-bounds for certain models (highly uncommon for LWRs); *Sm149xsab* dataset incorrectly has 0s in many places; log file refers to *SANM* nodal method but should say *SENM*.
5063	MPACT	VTK and VTU edits do not work for BWR models.
5069	MPACT	MPACT cannot initialize an upper reflector unless it is 100% one material.
5073	MPACT	SCRAM may work incorrectly with transient checkpoint file.
5077	MPACT	Transient checkpoint file does not contain delayed power history.
5078	MPACT	XSs are incorrectly computed for isotopes that have only absorption data.
5129	MPACT	VTK outputs do not work with more than two moderator rings around fuel.
5130	MPACT	Subgroup-cell method does not work with more than two moderator rings. **WORKAROUND**: Use the regular subgroup method.
1591	CTF	The channel splitting feature of CTF enables users 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 could create mass in the system when using this feature for transient simulations.
1157	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.
1636	CTF	There is a bug in the way that the calculated critical heat flux is time-relaxed for cases that use the W-3 correlation. This bug affects only models using the W-3 correlation and only transient simulations. Steady-state cases will still arrive at the same answer as that obtained if this bug did not exist.
1579	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).

Issue #	Component	Issue
1611	CTF	For cases with reversed flow at the outlet, when using the outlet pressure boundary condition, and with outlet voids greater than 0.2, the outlet void boundary condition could become inconsistent and lead to the code crashing.
1780	CTF	Using an inlet pressure/outlet pressure boundary condition has been observed to lead to discontinuous mass flow rates in the inlet momentum cell.
1826	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.
1780	CTF	Pressure distribution is discontinuous at the inlet boundary when modeling cases with reversed flow.
1786	CTF	CTF-predicted axial temperature distribution is nonlinear for cases with a uniform axial power distribution when using a nonuniform axial mesh.
2079	CTF	The fine mesh renoding capability used for quench front tracking on fuel rod surfaces does not work.
2086	CTF	The interpolation performed between CHF and minimum film boiling points is not correct, leading to the incorrect calculation of transition boiling heat transfer.
2220	CTF	The code crashes when attempting to include a heated plate geometry in the model.
2245	CTF	Multisection models that use nonsequential channel indexing will not work.
2894	CTF	Models with reverse flow crash with IAPWS steam tables but not with ASME steam tables.
2290	CTF	Radiative heat transfer model input does not work.
3133	CTF	The code is very susceptible to roundoff error for long-running transients. It has been observed that some tests fail in different builds or on different machines due to roundoff error.
3918	CTF	Units of mass balance information written to the `mass.out` file are written in standard units when they should be SI.
4947	CTF	Models that have axially varying water rods with a smaller-than-nominal diameter at the inlet will have an incorrect inlet mass flow rate specified when the model is built from Xml2ctf.
4953	CTF	Models generated by Xml2ctf that contain spacer grids with a form loss of 0.0 will not work in CTF.
5142	CTF	A debug build might encounter a DBC failure when building a CTF model from the XML file using Xml2ctf if that XML file has been thermally expanded.
N/A	VERAIO	Canopy might not run on some of the most recent Linux distributions. This is based on feedback from various users. (VERAView.)
4053	VERAIO	User errors with multiline input cards can lead to unexpected behavior or errors.
5135	VERAIO	react2xml –help message differs from that in the documentation.
N/A	VERAShift	Out of memory errors will occur when using unique pins for various quarter- and full-core problems. **WORKAROUND**: Do not use unique pins (transfer only the fission source).
4069	VERAShift	VERAShift dies ungracefully when specifying a tally outside the reflecting boundary for quarter-core problems.
4120	VERAShift	Pin adjoint in VERAShift output is incorrect for two-loop B&W plants.

Issue #	Component	Issue
4737	VERAShift	VERAShift multistate p9 FW-CADIS vessel fluence calculation runs out of memory.
4756	VERAShift	Neutron core pads are missing in Shift geometry without a barrel present.
4760	VERAShift	Neutron pad locations between MPACT and Shift are inconsistent for full-core problems.
4991	VERAShift	VERAShift fluences too high by order of magnitude for some MPI builds
5076	MAMBA	CRUD mass is not preserved when number and position of axial planes are changed between cycles on fuel reshuffle due to nearest neighbor interpolation method use in MAMBA. **WORKAROUND**: Do not change axial mesh between cycles, or split only existing planes but do not change existing plane locations.
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 eight cores, then 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 with 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`) cannot be run as concurrent iterators.
N/A	Dakota	dprepro does not 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 the presence of interface ID.
N/A	Dakota	Separate work directories are not created for concurrent iterators.
N/A	Dakota	D-optimal sampling designs do not work reliably.
N/A	Dakota	Coliny COBYLA methods might not return optimal solution.
N/A	Dakota	The sequential hybrid method might only propagate solutions with explicit model specification. **WORKAROUND**: Use *model_pointer*.
N/A	Dakota	Initial points are ignored by surrogates built from imported data.
N/A	Dakota	Efficient global method might 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 might 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 specification has invalid keywords.

REFERENCES

[1] J. A. Turner, K. Clarno, M. Sieger, R. Bartlett, B. Collins, R. Pawlowski, R. Schmidt, and R. Summers. The Virtual Environment for Reactor Applications (VERA): Design and Architecture. *Journal of Computational Physics*, 326:544–568, 2016.

[2] B. Collins., S. Stimpson, B.W. Kelley, M.T.H. Young, B. Kochunas, A. Graham, E.W. Larsen, T. Downar, and A. Godfrey. Stability and Accuracy of 3D Neutron Transport Simulations Using the 2D/1D Method in MPACT. *Journal of Computational Physics*, 326:612–628, 2016.

[3] MPACT Development Team. MPACT Theory Manual Version 4.3. Technical Report ORNL/SPR-2021/2330, Oak Ridge National Laboratory, 2021.

[4] I. C. Gauld, G. Radulescu, G. Ilas, B. D. Murphy, M. L. Williams, and D. Wiarda. Isotopic Depletion and Decay Methods and Analysis Capabilities in SCALE. *Nuclear Technology*, 174:169, 2011.

[5] R. K. Salko, M. N. Avramova, A. Wysocki, B. Hizoum, A. Toptan, J. Hu, N. Porter, T. Blyth, C. Dances, A. Gomez, C Jernigan, J. Kelly, and A. Abarca. CTF Theory Manual: Version 4.3. Technical Report ORNL/SPR-2022/2494, Oak Ridge National Laboratory, 2022.

[6] R. Williamson et al. Multidimensional Multiphysics Simulation of Nuclear Fuel Behavior. *Nuclear Materials*, 423:149–163, 2012.

[7] T. M. Pandya, S. R. Johnson, T. M. Evans, G. G. Davidson, S. P. Hamilton, and A. T. Godfrey. Implementation, Capabilities, and Benchmarking of Shift, a Massively Parallel Monte Carlo Radiation Transport Code. *Journal of Computational Physics*, 308:239–272, 2015.

[8] W. Gurecky, B. Collins, A. Elliot, D. Andersson, J. Galloway, B. Kendrick, and B. Okhuysen. MAMBA v.2.0 Theory Manual. Technical Report ORNL/SPR-2022/2524, Oak Ridge National Laboratory, March 2022.

[9] S. Palmtag, A. Godfrey, M. Baird, and E. Walker. VERAIn User's Manual. Technical Report ORNL/SPR-2022/2509, Oak Ridge National Laboratory, 2022.

[10] A. Godfrey, R. Lee, T. Lange, and E. Walker. VERAView User's Manual. Technical Report ORNL/SPR-2022/2502, Oak Ridge National Laboratory, 2022.

[11] A. Godfrey. VERA Core Physics Benchmark Progression Problem Specifications. Technical report, CASL-U-2012-0131-004, 2014.

[12] VERA Software Quality Assurance Plan. Technical Report VERA-QA-002 Rev 1, Oak Ridge National Laboratory, 2019.

[13] K. Kim, M. Williams, D. Wiarda, and A. Godfrey. Development of a New 47-Group Library for the CASL Neutronics Simulator. In *Proceedings of* ANS MC2015 – Joint International Conference on Mathematics and Computation (M&C), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method, 2015.

[14] B. M. Adams, L. E. Bauman, W. J. Bohnhoff, K. R. Dalbey, M. S. Ebeida, J. P. Eddy, M. S. Eldred, P. D. Hough, K. T. Hu, J. D. Jakeman, J. A. Stephens, L. P. Swiler, D. M. Vigil, and T. M. Wildey. Dakota, a Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.0 User's Manual. Technical Report SAND2014-4633, Sandia National Laboratory, 2014.

[15] A. Godfrey et al. VERAOut – VERA HDF5 Output Specification. Technical Report CASL-U-2014-0043-001, CASL, 2014.

[16] R. Lee, A. Godfrey, and E. Walker. VERAView Programmer's Manual. Technical Report ORNL/SPR-2022/2488, Oak Ridge National Laboratory, 2022.