



Getting Started with SCALE 6.2

April 2016

CONTENTS

	<u>Page</u>
INTRODUCTION.....	1
UPDATES IN SCALE 6.2	3
NUCLEAR DATA	3
ENDF/B-VII.1 CROSS-SECTION LIBRARIES.....	3
NEUTRON CROSS-SECTION COVARIANCE DATA.....	3
CONTINUOUS-ENERGY DATA PROCESSING.....	3
MULTIGROUP DATA PROCESSING.....	4
NUCLEAR DATA FILE FORMATS	5
MODERNIZED RESONANCE SELF-SHIELDING.....	5
CENTRM.....	7
BONDARENKO SELF-SHIELDING	7
GRAPHICAL USER INTERFACE – FULCRUM	8
CRITICALITY SAFETY	10
REDUCTION IN MEMORY REQUIREMENT OF CE INTERNAL STORAGE.....	10
FISSION SOURCE CONVERGENCE DIAGNOSTICS	11
PROBLEM-DEPENDENT DOPPLER BROADENING.....	11
DOPPLER BROADENED REJECTION CORRECTION	12
SOURCERER - HYBRID METHOD FOR STARTING SOURCE DISTRIBUTION.....	13
DISTRIBUTED MEMORY PARALLELISM VIA MPI	14
REACTION TALLIES	15
REACTOR PHYSICS	16
POLARIS – NEW ADVANCED LATTICE PHYSICS MODULE	16
CONTINUOUS-ENERGY MONTE CARLO DEPLETION	16
LATTICE PHYSICS WITH TRITON	17
BIAS REDUCTION FOR DEPLETION	17
RADIOACTIVE SOURCE TERM CHARACTERIZATION	18
ORIGEN	18
ORIGAMI – NEW ADVANCED USED FUEL ASSEMBLY CHARACTERIZATION.....	20
ORIGEN REACTOR LIBRARIES.....	20
ORIGAMI AUTOMATION TOOL.....	22
RADIATION SHIELDING	23
CONTINUOUS-ENERGY SHIELDING CALCULATIONS	23
SENSITIVITY AND UNCERTAINTY ANALYSIS	24
CE EIGENVALUE SENSITIVITY ANALYSIS.....	24
SAMPLER.....	26
OTHER UPDATES	28
KNOWN ISSUES.....	28
SYSTEM REQUIREMENTS	29
SYSTEM ARCHITECTURE:	29
SYSTEM REQUIREMENTS:	29
JAVA REQUIREMENTS:	29
INSTALLATION INSTRUCTIONS	30
SCALE 6.2 DATA.....	39

RUNNING SCALE	45
RUNNING SCALE FROM FULCRUM.....	45
WINDOWS.....	45
LINUX.....	45
MAC	45
RUNNING SCALE FROM THE COMMAND LINE	46
WINDOWS.....	46
LINUX.....	46
MAC	46
EXAMPLE INVOCATION	48
SCALE VARIABLES	49
PARALLEL EXECUTION CAPABILITY	52
SCALE SAMPLE PROBLEMS.....	54
BUILD INSTRUCTIONS	56
OVERVIEW.....	56
REQUIRED RESOURCES.....	56
MAC OSX RESOURCES.....	57
CMAKE CONFIGURATION.....	58
RECOMMENDED CONFIGURATION PROCEDURE.....	58
LINUX AND MAC CONFIGURATION	58
• MAKE BUILD DIRECTORY	58
• CMAKE INITIALIZATION.....	58
• CREATE YOUR CONFIGURATION.....	59
WINDOWS CONFIGURATION.....	59
COMPILATION.....	59
COMPILATION FLAGS	60
INSTALLATION.....	60
REFERENCES.....	61

List of Figures

	<u>Page</u>
Figure 1. SCALE calculated results for International Criticality Safety Benchmark Evaluation Project (ICSBEP) thermal mixed oxide critical systems.....	4
Figure 2. k_{eff} values for IEU-MET-FAST-005-001 with different MG options.....	7
Figure 3. SCALE Graphical User Interface – Fulcrum.....	9
Figure 4. Eigenvalues computed for a PWR pin cell with different temperature treatments.....	12
Figure 5. Fraction of failure to agree with the reference k_{eff} value for KENO calculations with different starting sources (Figure 4 from Ref. 9).....	14
Figure 6. Speedup for parallel KENO-VI calculations for a graphite-moderated reactor model.	15
Figure 7. Comparison of calculation times for 1470 TRITON calculations.....	17
Figure 8. Comparison of computational biases as a function of burnup.....	18
Figure 9. ORIGAMI Automator.....	22
Figure 10. MIX-COMP-THERM-004-001 total nuclide sensitivity Figure of Merit comparison.....	25
Figure 11. Uncertainty in plutonium isotopics as function of burnup.	27
Figure 12. NSIS installer loading.....	30
Figure 13. NSIS “Welcome” dialog.....	30
Figure 14. NSIS SCALE License Agreement.....	31
Figure 15. NSIS Choose Install Location.	32
Figure 16. NSIS Choose Start Menu Folder.	32
Figure 17. NSIS progress dialog.....	33
Figure 18. NSIS complete dialog.....	33
Figure 19. Mac OSX opening dialog.....	34
Figure 20. Mac OSX bundle contents.....	34
Figure 21. Linux Tar GZip.....	35
Figure 22. Windows Extract SCALE 6.2 source zip file.....	35
Figure 23. Windows SCALE 6.2 source directory.	36
Figure 24. Windows Extract SCALE 6.2 source directory contents.	37
Figure 25. Mac SCALE 6.2 source directory prior to extracting source code.....	38
Figure 26. SCALE 6.2 data installer welcome dialog.	39
Figure 27. SCALE license agreement.....	40
Figure 28. Setting typical data installation path on Windows.	41
Figure 29. Setting a typical data installation path on Linux.....	41
Figure 30. Setting a typical data installation path on Mac.....	42
Figure 31. Data installer directory creation error message.	42
Figure 32. Directory overwrite dialog.	43
Figure 33. Data selection dialog.	43
Figure 34. Media selection dialog.....	44
Figure 35. Windows desktop SCALE-6.2 shortcuts folder.	45
Figure 36. Mac SCALE-6.2 app in the Applications folder.	46
Figure 35. Sample problem output messages.	55

List of Tables

	<u>Page</u>
Table 1. SCALE Sequence Modernization.....	6
Table 2. Effect of DBRC on k_{inf} for a LWR fuel pin at 1200K	13
Table 3. Assembly and Lattices Types with Available ORIGEN Reactor Libraries.....	21
Table 4. Additional Information for ORIGEN Reactor Libraries.....	21
Table 5. MIX-COMP-THERM-004-001 total nuclide sensitivity coefficient comparison.	25
Table 6. Sensitivity method memory usage.....	26

Introduction

The SCALE Code System is a widely-used modeling and simulation suite for nuclear safety analysis and design that is developed, maintained, tested, and managed by the Reactor and Nuclear Systems Division (RNSD) of Oak Ridge National Laboratory (ORNL). SCALE provides a comprehensive, verified and validated, user-friendly tool set for criticality safety, reactor physics, radiation shielding, radioactive source term characterization, and sensitivity and uncertainty analysis. Since 1980, regulators, licensees, and research institutions around the world have used SCALE for safety analysis and design. SCALE provides an integrated framework with dozens of computational modules including 3 deterministic and 3 Monte Carlo radiation transport solvers that are selected based on the desired solution strategy. SCALE includes current nuclear data libraries and problem-dependent processing tools for continuous-energy (CE) and multigroup (MG) neutronics and coupled neutron-gamma calculations, as well as activation, depletion, and decay calculations. SCALE includes unique capabilities for automated variance reduction for shielding calculations, as well as sensitivity and uncertainty analysis. SCALE's graphical user interfaces assist with accurate system modeling, visualization of nuclear data, and convenient access to desired results. SCALE 6.2 represents one of the most comprehensive revisions in the history of SCALE, providing several new capabilities and significant improvements in many existing features.

New capabilities include:

- ENDF/B-VII.1 nuclear data libraries CE and MG with enhanced group structures,
- Neutron covariance data based on ENDF/B-VII.1 and supplemented with ORNL data,
- Covariance data for fission product yields and decay constants,
- Stochastic uncertainty and correlation quantification for any SCALE sequence with Sampler,
- Parallel calculations with KENO,
- Problem-dependent temperature corrections for CE calculations,
- CE shielding and criticality accident alarm system analysis with MAVRIC,
- CE depletion with T5-DEPL/T6-DEPL,
- CE sensitivity/uncertainty analysis with TSUNAMI-3D,
- Simplified and efficient LWR lattice physics with Polaris,
- Simplified spent fuel characterization with ORIGAMI and ORIGAMI Automator,
- Advanced fission source convergence acceleration capabilities with Sourcerer,
- Nuclear data library generation with AMPX,
- Integrated user interface with Fulcrum, and
- Many other new features.

Enhanced capabilities include:

- Accurate and efficient CE Monte Carlo methods for eigenvalue and fixed source calculations,
- Improved MG resonance self-shielding methodologies and data,
- Resonance self-shielding with modernized and efficient XSPROC integrated into most sequences,

- Accelerated calculations with TRITON (generally 4x faster than SCALE 6.1),
- Spent fuel characterization with 1470 new reactor-specific libraries for ORIGEN,
- Keyword input for ORIGEN,
- Extension of the maximum mixture number to values well beyond the previous limit of 2147 to ~2 billion,
- Nuclear data formats enabling the use of more than 999 energy groups,
- Updated standard composition library to provide more accurate use of natural abundances, and
- Numerous other enhancements for improved usability and stability.

The user documentation for SCALE has also been substantially updated and reorganized around capabilities instead of the historical division of function modules, control modules, etc.

This getting started guide provides an overview of many of the updated features of SCALE 6.2, instructions on how to install SCALE, instructions on using all the features of the SCALE runtime environment command line interface, and instructions on how to build SCALE. The SCALE user manual provides comprehensive documentation of all computational capabilities, nuclear data, input requirements, and output edits. Additional resources are available in the SCALE primers, located in the *docs/primers* directory after installation. These primers provide step-by-step instructions for running SCALE. However, at the time of release, not all primers have been updated to reflect the latest features and user interface of SCALE 6.2. Additional resources are available on the SCALE website at <http://scale.ornl.gov> or by e-mailing scalehelp@ornl.gov.

Updates in SCALE 6.2

Nuclear Data

ENDF/B-VII.1 Cross-section Libraries

ENDF/B-VII.1 nuclear data libraries are introduced in SCALE 6.2. CE data are available for general-purpose neutron, gamma, and coupled neutron/gamma calculations. MG neutron libraries in the 252- and 56-group structure are available, where the 252-group library is for general-purpose applications and the 56-group library is intended for light water reactor analysis. Coupled neutron/gamma MG libraries are available in a fine 200-neutron/47-gamma group structure and a broad 28-neutron/19-gamma structure. An additional 8-group neutron library is available for code testing purposes but is not intended for production use.

Neutron Cross-section Covariance Data

Updated cross-section covariance libraries are provided with SCALE 6.2 for use with the sensitivity and uncertainty modules. The data has been assembled from a variety sources, including high-fidelity covariance evaluations from ENDF/B-VII.1, other domestic and international evaluations, as well as approximate uncertainties obtained from a collaborative project performed by Brookhaven National Laboratory, Los Alamos National Laboratory, and ORNL. In addition, the covariance library now uses a 56-group structure for broad group analysis, which is suitable for most applications, as well as the 252-group structure for fine group analysis, such as for energy-dependent reaction rates. These libraries are generated for compatibility with the ENDF/B-VII.1 56-group lattice physics library, 252-group criticality library, and continuous-energy analysis. The current SCALE covariance library spans the full energy range of the multigroup cross-section libraries. The new 56-group and 252-group covariance libraries (56groupcov7.1 and 252groupcov7.1) are recommended for all applications. However, the previous library (44groupcov) distributed with SCALE 6.0 and SCALE 6.1 is retained for backwards compatibility. Covariance data are available for 456 materials, including some duplication for materials with multiple thermal scattering kernels. The SCALE 6.2 56-group covariance library (56groupcov7.1) is the default library for uncertainty calculations, but other libraries can be accessed by setting the appropriate parameter in the sensitivity/uncertainty analysis codes.

Continuous-energy Data Processing

Investigations into the CE data generated by the AMPX code system for deployment in SCALE 6.0 and 6.1 revealed a need for improvement in the $S(\alpha,\beta)$ treatment, especially for forward-peaked kinematics. The SCALE 6.2 ENDF/B-VII.0 and ENDF/B-VII.1 data libraries have been generated using new AMPX processing procedures, and the benchmark testing results with SCALE 6.2 show substantially improved results relative to SCALE 6.0 and 6.1. Select critical benchmark results for thermal mixed oxide (MOX) systems are provided in Figure 1, which shows that the bias in the SCALE 6.2 ENDF/B-VII.0 CE results is reduced relative to SCALE 6.1. Additional testing has revealed that biases for burned light water reactor (LWR) fuel as large as 1000 pcm are

resolved with the improved treatment. Additional details and further results are provided in Ref. 1.

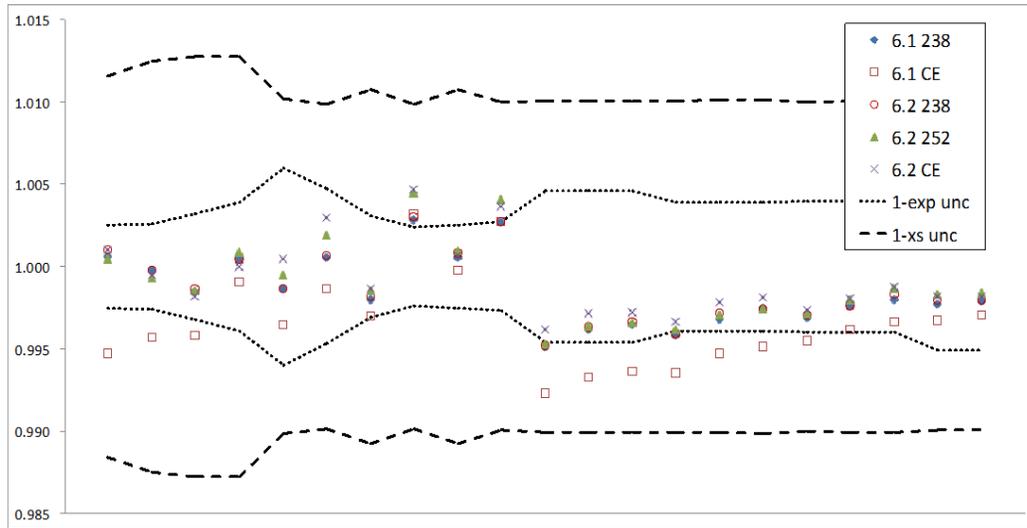


Figure 1. SCALe calculated results for International Criticality Safety Benchmark Evaluation Project (ICSBEP) thermal mixed oxide critical systems.

Additionally, the probability tables that provide CE treatment in the unresolved resonance range have been improved, primarily through the inclusion of additional resolution and error correction. Testing with the new probability tables has shown reduced biases for systems that are sensitive to the intermediate energy range.¹

Historically, only neutron CE data libraries with a specific reaction subset of the ENDF libraries have been supported in KENO Monte Carlo criticality safety codes. To support CE depletion, sensitivity analysis, and coupled neutron-gamma shielding analysis, AMPX capabilities for the generation of CE neutron data have been improved and extended. Capabilities were added to generate gamma interaction data and to produce gamma yield data from neutron interactions. In addition, wide ranges of neutron interactions were also added in order to support creating reaction responses in Monaco.

The improved CE data has been comprehensively reviewed and verified and validated with approximately 5000 infinite medium eigenvalue tests, 6300 fixed-source transmission tests, and 400 criticality and shielding benchmark experiments to ensure robust and accurate calculations.

Multigroup Data Processing

For MG analyses, ORNL has performed detailed comparisons between SCALe CE and MG results, and historically a bias of 200-500 pcm has been observed where the 238-group library is applied to LWR systems. The new 252-group structure for SCALe 6.2 provides a more detailed representation of the ²³⁸U resonance structure. AMPX has been used to develop a new 56- and 252-energy-group ENDF/B-VII.1 neutron cross-section libraries. In addition to the new group structures, these libraries have been

generated using a new weighting spectrum with improved resonance self-shielding parameters. Specifically, for actinides and hydrogen in H₂O, a temperature-dependent CE flux spectrum has been generated with the CE CENTRM module for a PWR pin cell, and the CE flux spectrum has been used as a weighting function for library generation. Intermediate resonance parameters (lambdas) for all isotopes have been included in the library, which provides a capability for improved self-shielding with the Bondarenko method and Polaris. Bondarenko shielding factors, as a function of background cross section and temperature, were computed for all actinides (except ²³⁵U, ²³⁸U, ²³⁹Pu, ²⁴⁰Pu, and ²⁴¹Pu) using CE spectra calculated by CENTRM for homogeneous mixtures of the resonance material and hydrogen, corresponding to the respective background cross section. For the nuclides ²³⁵U, ²³⁸U, ²³⁹Pu, ²⁴⁰Pu, and ²⁴¹Pu, the Bondarenko factors are based on CENTRM CE calculations for heterogeneous pin cell models that span the range of anticipated self-shielding conditions. The thermal cutoff for moderators and the free-gas approximation has been raised from 3 eV to 5 eV. The new library has been tested by analyzing a wide variety of critical benchmark experiments and by comparing results with CE Monte Carlo results. The 252-group results presented in Figure 1 demonstrate consistent performance with the SCALE 6.2 CE results for the thermal MOX critical benchmark experiments. Based on additional studies with the 252-group library, computational benchmark comparisons with CE results at room temperature and elevated temperatures show agreement within 100 pcm in most cases.

Nuclear Data File Formats

A new binary data format is introduced to SCALE and applied to modernize the format of the AMPX multigroup cross section data libraries, replacing the AMPX Working and Master formats that were designed in the 1960s. The previous formats were restricting the extension of capabilities desired to provide improved physics.

BOFF (Binary Object Formatted File) is a binary data format designed for flexibility and compactness and is intended to compete with native binary file formats that incorporate optimization tricks implicit to the developer and the data being stored. BOFF revolves around the steadfast data structures of primitive, array, and object data and can provide backward and forward data format compatibility. BOFF provides the capability to store data in a hierarchical manner utilizing objects, arrays, and keyed-values. The BOFF format is implemented in the AMPX multigroup format to provide the ability to incorporate additional data in future updates.

Modernized Resonance Self-Shielding

Several SCALE sequences have been updated to perform resonance self-shielding of multigroup cross sections and one-dimensional neutron transport calculations using the modern XSPROC module. XSPROC provides the capabilities of BONAMI, CENTRM, PMC, WORKER, and XSDRN and has been demonstrated to produce equivalent results to the prior independent codes. With XSPROC runtime and memory requirements are substantially improved, especially when performing calculations with many unit cells. Generally speedups in self shielding of about 3x are realized, but in an extreme test case using an as-loaded spent nuclear fuel storage package with hundreds of unique materials, the cross section processing and cell homogenization time was reduced from 5 days to

3.7 minutes, while still obtaining equivalent results. XSProc is only partially implemented across most SCALE sequences, as shown in Table 1, where modernized sequences with XSProc and legacy sequences that run that standalone codes are identified.

Table 1. SCALE Sequence Modernization

Sequence	Modern	Legacy
CSAS-MG	✓	
CSAS1	✓	
CSAS1X	✓	
CSAS5	✓	
CSAS6	✓	
CSASI	✓	
CSASIX	✓	
MAVRIC	✓	
STARBUCS		✓
T-NEWT	✓	
T-XSDRN	✓	
T-XSEC	✓	
T5-DEPL	✓	
T6-DEPL	✓	
T-DEPL	✓	
T-DEPL-1D	✓	
TSUNAMI-1D	✓	
TSUNAMI-2D		✓
TSUNAMI-3D_K5	✓	
TSUNAMI-3D_K6	✓	

XSProc provides for infinite homogeneous medium, lattice cell, multiregion, and double heterogeneous cell types. For all but the double heterogeneous cells, a new capability is implemented for accelerated self-shielding using intermediate resonance (IR) approximation Bondarenko treatment for the full energy range (PARM=BONAMI) as well as the more rigorous option of continuous-energy treatment in the resolved resonance range (PARM=CENTRM), with default treatment set as continuous-energy. Only the CENTRM path is available for double heterogeneous cell types.

XSProc provides capabilities for:

- Resonance self-shielding of microscopic data,
- Macroscopic cross sections for mixtures,
- One-dimension multigroup transport calculations to calculate eigenvalues and flux weighting functions,
- Group collapsing of cross sections using flux spectra from the one-dimensional eigenvalue calculation or user input fixed source spectra, and
- Spatial homogenization of cross sections across material zones.

Sequence-specific control for the generation of on-disk or in-memory data libraries for microscopic or macroscopic data containing selected reaction cross sections and scattering data is available with this modernized tool.

CENTRM

A new two-dimensional CENTRM option is available to explicitly treat the boundary of square-pitched and triangular-pitched light water reactor fuel pins with a Method-of-Characteristics (MoC) CE transport option, resulting in approximately 100 pcm reduction in bias between MG and CE calculations. This is now the default option for fuel lattice calculations using CENTRM through XSPROC within SCALE sequences.

New options are available in CENTRM/PMC to address the effects of resonance self-shielding on scattering matrices, using the $N2D=$ option in *centrm data* in *celldata*. $N2D=2$ performs self-shielding corrections on the 0th order scattering data (like default the $N2D=-1$ method) but also corrects the higher order Legendre moments with corresponding higher order fluxes. $N2D=-2$ similarly treats the higher order Legendre moments and explicitly treats within-group elastic scattering removal. The impact of these options is demonstrated on the IEU-MET-FAST-005-001 fast system with a steel reflector with k_{eff} results shown in Figure 2. Here it is observed that the use of the $N2D=2$ option produces MG results that are more consistent with the reference CE Monte Carlo results. However, these results are not universal for all calculations, so the $N2D=-2$ option should be applied after careful investigation of each system type. In particular, the use of this option for systems with large reflectors is not recommended.

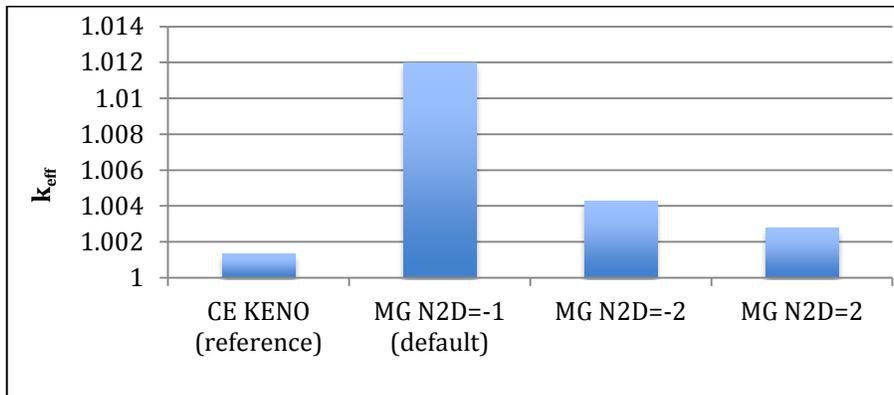


Figure 2. k_{eff} values for IEU-MET-FAST-005-001 with different MG options.

Bondarenko Self-Shielding

BONAMI has been rewritten as part of the overall SCALE modernization effort. BONAMI now implements IR approximation theory as well as the previous narrow resonance (NR) approximation. The IR approximation allows for improved accuracy for systems with overlapping resonances and provides a rapid self-shielding option using only full-range Bondarenko factors. This technique produces accurate results and provides up to a 10x speedup relative to CE treatment with CENTRM. IR theory is

described in the updated BONAMI section, Section 8.2. To enable this option set *PARM=BONAMI* on the sequence specification record (e.g. *=t-depl PARM=BONAMI*) and *IROPT=1* in the *more data* section of *celldata*.

Graphical User Interface – Fulcrum

Fulcrum is a cross platform graphical user interface designed to create, edit, validate and visualize SCALE input, output, and data files. Historically, SCALE has provided several special-purpose graphical user interfaces, which operate only on specific platforms and are loosely integrated with SCALE's computational and data components. Fulcrum, in contrast, is intended to provide a single user interface that directly integrates with SCALE's internal resources to provide a consistent experience between Fulcrum and SCALE's command line interface.

The concept of Fulcrum is based on decades of feedback from our user community through the release of numerous interfaces. In contrast to the SCALE 6.1 GeeWiz interface with many layers of dialog boxes, Fulcrum directly connects the user with the text form of the input file, while providing inline features to assist with building correct inputs. Fulcrum provides input editing and navigation, interactive geometry visualization for KENO V.a, KENO-VI and NEWT, job execution, overlay of mesh results within a geometry view, and plotting of data from most SCALE file formats. An error checker interactively identifies poorly formed input with spelling errors or data entry omissions for all SCALE sequences. The Hierarchical Input Validation Engine (HIVE) identifies allowed data ranges and interdependencies in the input and reports inconsistencies to the user. Fulcrum interactively processes standard composition data to produce a mixing table, lists expanded input aliases for review, provides an internal listing of input as is required for Sampler material and geometry perturbation analysis, and launches the SCALE sample problems.

The layout of panels in Fulcrum is highly configurable to accommodate the preferences of many users, as shown in Figure 3.

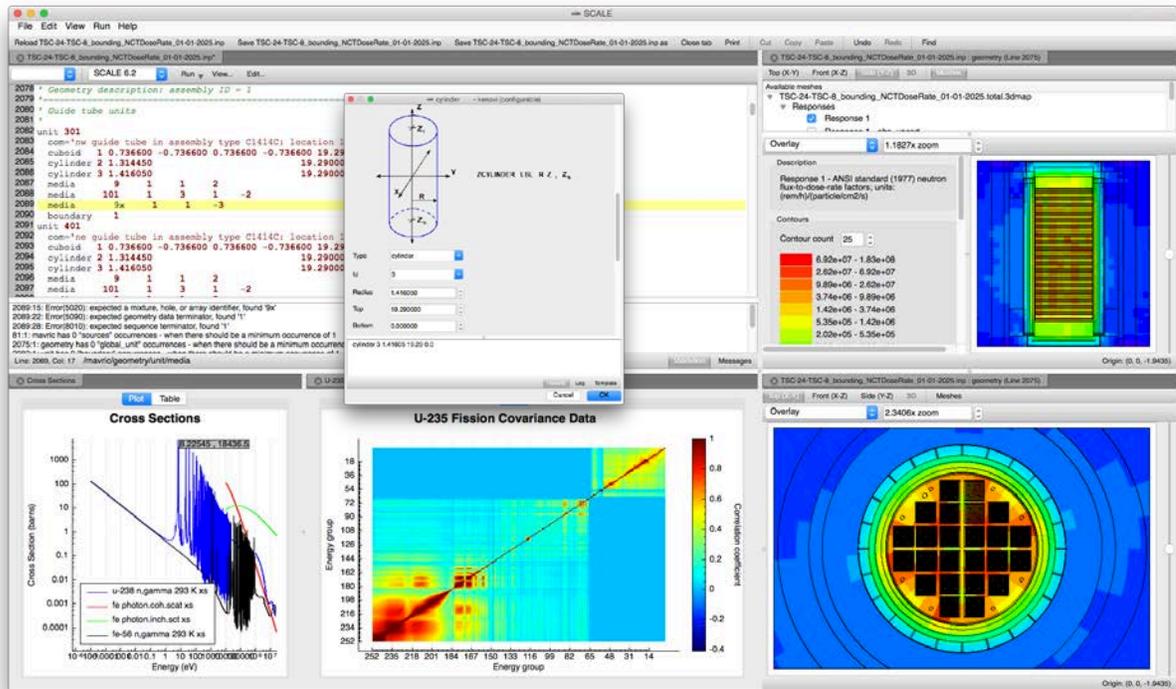


Figure 3. SCALE Graphical User Interface – Fulcrum.

The Fulcrum user interface provides numerous features, such as:

- Operates on all supported platforms, Linux, Mac and Windows;
- Can edit, view, and run multiple inputs simultaneously;
- Context aware auto-completion input generation menus are presented when *ctrl+space* is pressed within an input file,
- Graphical forms-based input is available to assist users with the creation of sequence blocks and specific input components (press *ctrl+space* and select a *configurable* item);
- Input is validated as it is entered, and erroneous or missing input is identified in the *Validation* panel;
- Geometry models can be visualized for sequences that use KENO V.a, KENO-VI, Monaco, and NEWT (*View...* menu shown in toolbar associated with relevant input file);
- Fulcrum is pre-configured to run SCALE 6.2, but can be custom configured for other codes;
- Associated files (such as *.out*, *.msg*, *plot*, or data files) are shown in contextual pop-up menu for each for file in the navigation panel and can be opened (right-click on input file name in navigation panel);
- Associated files not supported in Fulcrum open a local tool for appropriate viewing (e.g. *.html* files launch the system web browser);
- Mesh tallies stored in *.3dmap* files can be overlaid with geometry view (*Meshes* on toolbar in geometry view, right click in Available Meshes panel to load a *.3dmap* file).
- Data plotting is available for the following SCALE files:

- Multigroup and continuous-energy cross section data;
- Cross section covariance data;
- ORIGEN gamma data;
- UNF-STANDARDS time series;
- ORIGEN concentration file (*f71*) with integrated unit conversion (OPUS capability);
- Sensitivity data file (*.sdf*);
- KENO reaction rate and flux file (*.kmt*);
- Ptolemy plot file (*.ptp*) (previously Javapeño 2D plot, *.plt*);
- MAVRIC Chart Plot file (*.chart*).
- Data files with file extensions (e.g. *.sdf*, *.kmt*, etc.) are access through *File/Open file...*;
- Other file types, especially SCALE nuclear data files, have a specific menu item in the *File* menu (e.g. *Open continuous-energy library...*);
- Plots can now be saved in an editable format as a SCALE Plot File (*.spf*);
- Data tables are available for data shown in a plot;
- Geometry volumes can be computed for KENO and Monaco models;
- Online help is available; as well as
- Too many features to list here.

As Fulcrum is a new user interface, users are directed to the *Help* menu within the application itself for *Help documentation*.

Criticality Safety

KENO has been substantially improved for SCALE 6.2, especially for the accuracy and efficiency of CE calculations, CE temperature corrections, source convergence diagnostics and acceleration, and parallel capabilities.

Reduction in Memory Requirement of CE Internal Storage

Numerous improvements were made to enhance the CE capabilities of KENO, especially to reduce the memory requirements of the calculations. CE calculations now require 40-99% less memory than previous versions with no loss of accuracy in the results. Improved memory efficiency will be observed for all calculations, but especially for models with many materials and/or temperatures. Models that previously would have required 100s of GB of memory with SCALE 6.0 and 6.1 can now be performed with only a few GB of memory.

User controllable options:

- *UUM*: Optional *unionization* of mixture-dependent cross section data results in a 50-90% reduction in memory requirements for a single material, with no additional memory required for the use of the same nuclide in more than one mixture. Previous burned fuel calculations that previously would have required 100s of GB of memory can now be run with just a few GB through the use of the default setting of *UUM=no* in the KENO parameter data.

- *M2U*: Optional nuclide level energy grid unionization (map2union) controls the unionization of all reaction types within a single nuclide. The default behavior, *M2U=yes*, in KENO parameter data helps to reduce runtime and was the default behavior in SCALE 6.0 and 6.1. Optionally disabling map2union (*M2U=no*) results in ~20% reduction in memory with ~10% increase in runtime.

Other internal optimizations include:

- Removed redundant copies of temperature independent data to improve memory efficiency for models with multiple temperatures. Previously, each set of temperature dependent cross sections also included a redundant copy of the temperature independent data, which caused an almost linear increase in memory for each new temperature. With this update, the temperature independent data are shared by all temperatures, resulting in >50% memory savings for the addition of each temperature relative to the techniques implemented in SCALE 6.0 and 6.1.
- Kinematics (scattering) data structures were updated to optimize data storage, especially with an updated structure in the CE data itself. These updates result in ~40% reduction in memory requirements using the same runtime.
- Changed internal data storage precision from double precision to single precision where possible, resulting in a 15-45% reduction in memory requirements, depending on the nuclides used in the model.
- Introduced optional data loading by filtering data during reading instead of loading all available data whether it is needed or not. Filtering methods such as energy range, reaction types, data types, etc., results in ~20% reduction in memory requirements.

Fission Source Convergence Diagnostics

Prior to SCALE 6.2, KENO provided only plots of k_{eff} by generation and average k_{eff} for visual inspection of source convergence, followed by a χ^2 statistical assessment of convergence. With SCALE 6.2, Shannon Entropy fission source convergence diagnostic techniques have been implemented in KENO to provide improved confidence in the computed results, as well reduced simulation times in some cases. Confirming the convergence of the fission source distribution is especially useful to avoid the false convergence of k_{eff} and neutron flux tallies that can be caused by insufficient sampling of important portions of the system.² Source convergence diagnostics are enabled with *SCD=yes* (default) in the KENO parameter data.

Problem-Dependent Doppler Broadening

The CE data libraries distributed with SCALE are provided with only about five temperatures per isotope. The Doppler broadening temperature corrections using only a few temperatures may not match the desired temperature of the calculation. When temperatures of the KENO model are different from those present on the library, KENO selects the nearest temperature, which can be several hundred degrees from the desired

temperature, producing results that can vary significantly from those that would be produced at the correct temperature.

New methods have been developed and implemented to provide *problem-dependent* temperature corrections by Doppler broadening the pointwise data in the resolved resonance region and the probability tables in the unresolved resonance region when the cross sections are loaded for the calculation.³ The thermal scattering data are also updated to requested temperature. The runtime penalty for this methodology is negligible, as all temperature corrections are performed as the calculation begins, typically requiring only a few seconds to a few minutes depending on the number of nuclides and temperatures used.

The eigenvalues computed for a typical fresh pressurized water reactor (PWR) pin cell using the nearest selected CE temperature, and problem-dependent CE temperature treatments are shown in Figure 4. Problem-dependent Doppler broadening is controlled with $DBX=0$ to select the nearest temperature, $DBX=1$ to perform problem-dependent corrections for the resolved and unresolved resonance ranges, and $DBX=2$ (default) to perform also corrections for the $S(\alpha,\beta)$ thermal scattering data.

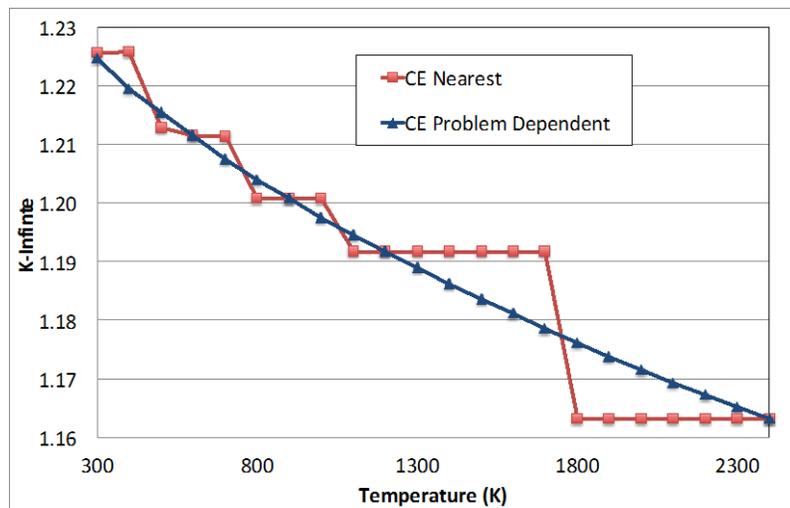


Figure 4. Eigenvalues computed for a PWR pin cell with different temperature treatments.

Doppler Broadened Rejection Correction

The implementation of Doppler Broadened Rejection Correction (DBRC) techniques provides further enhancements for calculations with elevated temperatures.⁴ As shown in Table 2, DBRC in KENO presents a reactivity correction of approximately 300 pcm relative to the default methodology for a 1200K LWR fuel pin, consistent with that predicted with MCNPX by the originators of the methodology.⁵ DBRC is enabled with

$DBR=$ in the KENO parameter data. Available options are $DBR=0$ to disable the correction (default), $DBR=1$ to perform DBRC for ^{238}U only, and $DBR=2$ to perform DBRC for all major actinides. The use of DBRC only impacts calculations at elevated temperatures, and there is a runtime penalty for the use of this methodology. As such, it is not recommended for calculations near room temperature and

Table 2. Effect of DBRC on k_{inf} for a LWR fuel pin at 1200K

Case	Default	DBRC	Difference (pcm)
MCNPX	1.31137 +/- 9E-5	1.30791 +/- 9E-5	-346
KENO-VI	1.31029 +/- 15E-5	1.30730 +/- 15E-5	-299

Sourcerer - Hybrid Method for Starting Source Distribution

The *Sourcerer* sequence introduced in SCALE 6.2 uses the Denovo⁶ discrete-ordinates code to generate a starting fission source distribution in a KENO Monte Carlo calculation. Initial studies^{7,8} have shown that using a starting fission distribution that is similar to the true fission distribution can both reduce the number of skipped generations required for fission source convergence and significantly improve the reliability of the final k_{eff} result.

For many criticality safety applications, the additional step of performing a deterministic calculation to initialize the starting fission source distribution is not necessary. However, for the most challenging criticality safety analyses, such as spent nuclear fuel loaded transportation packages with a mixed loading of low- and high-burnup fuel, even a low-fidelity deterministic solution for the fission source produces more reliable results than the typical starting distributions of uniform or cosine functions over the fissionable regions, as demonstrated in a recent study.⁹ In that study, a cask holding 24 spent fuel assemblies was examined using a uniformly distributed starting source and a deterministically calculated starting source. Multiple clones of KENO were run (with different random number seeds) for different values of skipped cycles. The number of clones that gave an incorrect result for k_{eff} was then tabulated. The results from that study, presented in Figure 5, show that using a deterministic starting source significantly increases the k_{eff} reliability. See the *Sourcerer* documentation, Section 2.4 for full details.

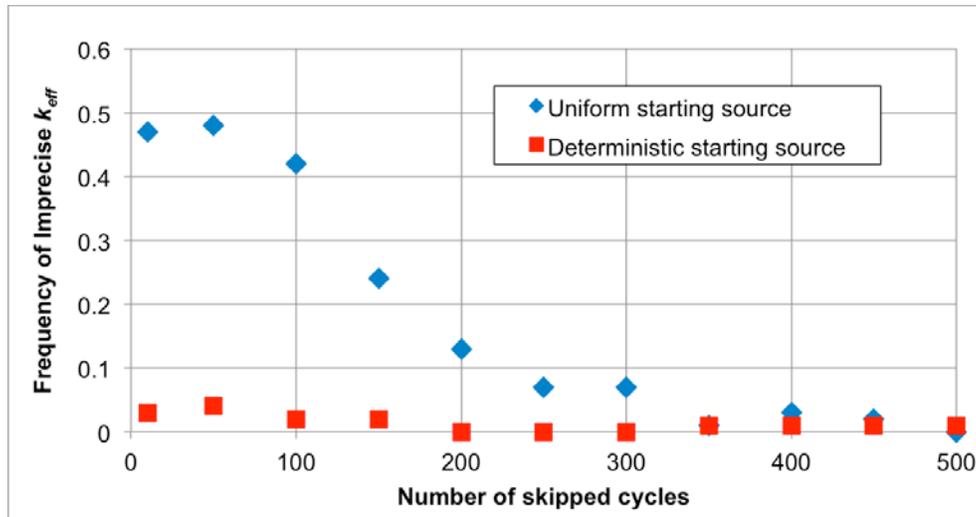


Figure 5. Fraction of failure to agree with the reference k_{eff} value for KENO calculations with different starting sources (Figure 4 from Ref. 9).

Distributed Memory Parallelism via MPI

In addition to the numerous improvements described that directly affect solution accuracy and efficiency, parallel computation capabilities, especially for Linux clusters, have been added to KENO to provide reductions in wall clock time, especially for S/U analysis or Monte Carlo depletion on computer clusters. By introducing a simple master-slave approach via Message Passing Interface (MPI),¹⁰ KENO runs different random walks concurrently on the replicated geometry within the same generation. The fission source and other tallied quantities are gathered at the end of each generation by the master process; then, they are processed either for final edits or next generations.

The parallel performance of KENO as used in a CE calculation for a graphite-moderated reactor model is shown in Figure 6. These tests were conducted on a heterogeneous Linux cluster where the size of the nodes varies from 4 to 16 cores with differing processor speeds, much like SCALE users may encounter in practice. Tests were conducted with systematically increasing numbers of particles per generation, and various combinations of options were enabled to develop the distributions of speedups for each number of MPI processes shown in the figure. With larger numbers of particles per generation, KENO provides nearly linear speedup on the 64 processors tested here and has been successfully demonstrated on hundreds of processors.

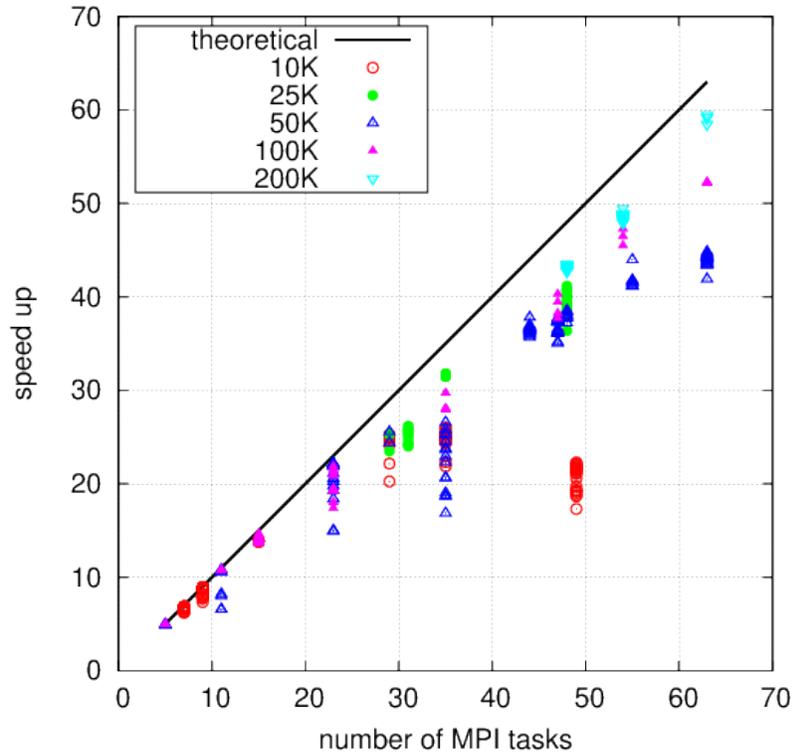


Figure 6. Speedup for parallel KENO-VI calculations for a graphite-moderated reactor model.

Reaction Tallies

User configurable reaction rate tallies are now available within KENO CE calculations. The tallies are specified in the *REACTION* block of KENO.

Reactor Physics

Polaris – New Advanced Lattice Physics Module

Polaris is a new 2D lattice physics code for SCALE that utilizes a new MG self-shielding method called the Embedded Self-Shielding Method (ESSM)¹¹ and a new MoC transport solver. Polaris provides simplified input, where only a few lines are required to describe the entire model. Polaris provides substantially simplified input requirements and improved runtime performance compared to TRITON (see Polaris section, Section 3.2).

The ESSM approach computes MG self-shielded cross sections using Bondarenko interpolation. The background cross section utilized in the interpolation is determined by a series of 2D MoC fixed-source calculations, similar to the subgroup method that does not require explicit *celldata* input. Additionally, heterogeneous lattices are explicitly treated without the need to externally compute Dancoff factors. Additional details on ESSM are provided in Ref. 11.

Polaris currently employs ESSM with either the 252-group or 56-group ENDF/B-VII.1 libraries. Each library contains cross sections, IR parameters, and full-range Bondarenko factors for all nuclides.

Polaris utilizes the self-shielded cross sections within a MG 2D eigenvalue calculation using the new MoC transport solver. The MoC solver has been developed within Denovo, which was originally developed for parallel 3D Cartesian mesh MG discrete ordinates (S_N) calculations.⁶ Polaris also provides a critical spectrum calculation for correcting the flux distribution for computing both few-group homogenized cross-section edits and depletion reaction rates.

Polaris is integrated with ORIGEN for depletion calculations. The depletion of each pin, or radial subregion of the pin, is based on the local normalized flux distribution. Cross-section values in the ORIGEN transition matrix are updated from the MG self-shielded cross sections and the MG flux distribution for each depletion region. The critical-spectrum correction to the flux distribution for depletion is controlled by an input user option. The cross-section updates are performed in-memory as compared to the file-based approach utilized in the TRITON lattice physics sequence. Polaris supports branch calculations for the generation of few group constants for reactor core simulators.

Continuous-Energy Monte Carlo Depletion

SCALE 6.1 provided MG Monte Carlo depletion that coupled SCALE MG cross-section processing capabilities with KENO and ORIGEN. A new CE-based KENO/ORIGEN Monte Carlo depletion capability has been developed and can be utilized by simply changing the input library specification. CE depletion is especially useful for models with complex geometry that present difficulties in obtaining accurate resonance self-shielded MG data, and for models with many depletion regions where run-time to generate and store the resonance self-shielded cross-section data for each material is prohibitive.

Lattice Physics with TRITON

Two-dimensional lattice physics calculations with TRITON will realize substantial speedups due to the use of XSProc for resonance self-shielding as well as numerous optimizations within the NEWT code. Speedups of 2-6x faster calculations are common, and some models have realized speedups of 30x relative to previous versions of NEWT. Calculation times for 1470 TRITON calculations used to generate the ORIGEN reactor libraries for SCALE 6.1.3 and SCALE 6.2 are shown in Figure 7. For the calculation, the average calculation time was reduced by a factor of 4.

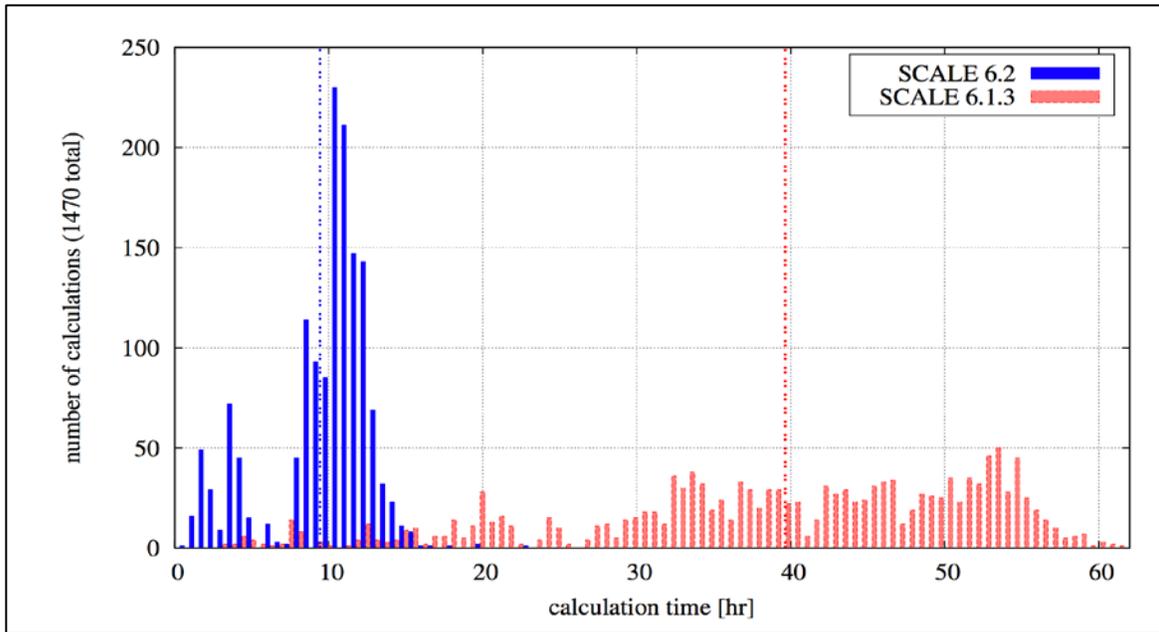


Figure 7. Comparison of calculation times for 1470 TRITON calculations.

A material SWAP capability is now available within the TIMETABLE block of TRITON. This feature allows a user to switch the material present during a depletion calculation and is especially useful for removable absorber materials.

The ASSIGN capability of TRITON is available for many calculations, where the same resonance self-shielding calculation can be applied to many similar materials. Unfortunately, ASSIGN is not yet implemented for BRANCH, TIMETABLE, or SWAP calculations.

Bias Reduction for Depletion

Historical SCALE biases have been mitigated as a result of the numerous enhancements included in SCALE 6.2.

Figure 8 shows the results for a reference PWR pin cell burnup calculation using SCALE 6.1 and SCALE 6.2. Here, the SCALE 6.2 continuous-energy results are the reference, and the SCALE 6.1 continuous-energy results are simply eigenvalue statepoints using the isotopics provide by the SCALE 6.2 results, as CE depletion is not

available in SCALE 6.1. The 1000 pcm bias in the SCALE 6.1 CE results are due to erroneous treatment of thermal scattering data in the AMPX CE data distributed with SCALE 6.1. The SCALE 6.1 TRITON 238 group results using the default settings from SCALE 6.1 demonstrate the typical 400-500 pcm bias observed for SCALE 6.1 LWR calculations. The TRITON 252 group results apply the improved nuclear data library, the CENTRM MoC solver, and the enhanced within group treatment for CENTRM/PMC, $N2D=-2$. The SCALE 6.2 Polaris results apply the default settings in Polaris with the ESSM methodology and IR factors on the library. These calculations demonstrate the impact of improvements in both the multigroup libraries and resonance self-shielding techniques as well as improvements in the continuous-energy treatments from SCALE 6.1 to SCALE 6.2.

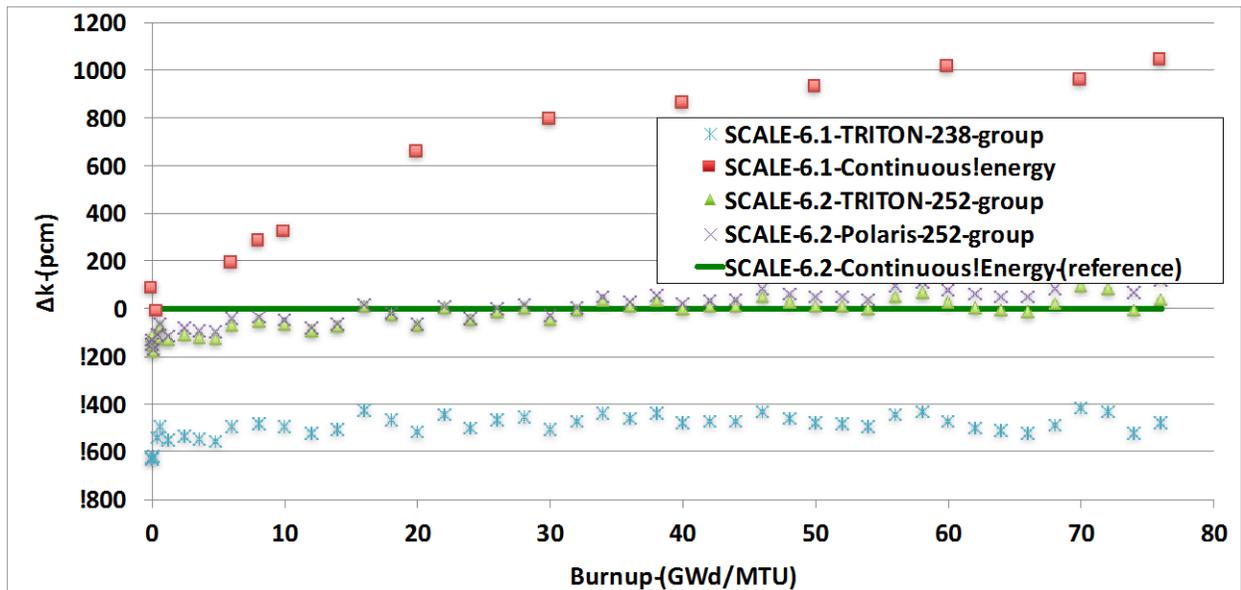


Figure 8. Comparison of computational biases as a function of burnup.

Radioactive Source Term Characterization

ORIGEN

SCALE includes the ORIGEN code and its comprehensive depletion, activation, decay, gamma-ray, and x-ray library with over 2,200 nuclides. ORIGEN and its nuclear data libraries have been updated to provide convenient modular interfaces within SCALE, and these interfaces also provide easy access to ORIGEN’s robust capabilities by other software packages.

The input for ORIGEN has been updated to provide a modern interface using the SCALE’s own Standard Object Notation (SON). The legacy FIDO input interface will continue to be supported for backwards compatibility, but users are encouraged to begin using the modern input through input files as well as the forms-based input in Fulcrum. An excerpt from an example problem is included below to show this new structured and keyword based input.

```

=origen
'three cycles of irradiation plus decay
bounds {neutron="xn27g19v7.0" gamma="xn27g19v7.0"}
case(c1){
  lib{ file="ft33f001" pos=1 }
  time=[8i 50 500]
  power=[10r40]
  mat{ iso=[u235=4e3 u238=960e3] }
}
case(c2){
  lib{ pos=2 }
  time=[8i 550 1000]
  power=[10r40]
}
case(c3){
  lib{ pos=3 }
  time=[8i 1050 1500]
  power=[10r40]
}
case(cool){
  time{ start=0 t=[20L 0.001 100] units=YEARS }
  save{ file="snf.f71" time_offset=1500 }
  gamma{ sublib=ALL brem_medium=UO2 }
  print{ neutron{ spectra=yes } }
  neutron{ alphan_medium=UO2 }
}
end

```

The ENDF/B-VII.0 based fission product yields used by ORIGEN are updated to provide improved agreement with burst fission experiments for ^{235}U , ^{238}U , ^{239}Pu , and ^{241}Pu . The changes correct inconsistencies that were introduced in ENDF/B when updating the nuclear decay data in ENDF/B VII.0 and not updating the fission product yields that are based on England and Rider (1994). These yield adjustments improve the consistency between the direct fission yields and decay data, and the cumulative yields in ENDF/B-VII.0.

ORIGEN has been enhanced to provide an alternative solver based on the Chebyshev Rational Approximation Method (CRAM). When compared to a robust reference solution for a depletion problem, the traditional matrix power series and Bateman solver predicts isotopic concentrations with an average relative error of about $10\text{E-}5$, where the CRAM solver produces an average relative error of about $10\text{E-}15$ for the same case. The CRAM solver can also handle essentially arbitrary step lengths through internal substepping with minimal computational cost, so the runtimes are generally faster for large depletion problems. The previously mentioned depletion case required 550 ms to execute 8 substeps with the traditional solver, where CRAM completed the more accurate calculation in 44 ms. However, CRAM is slower than the traditional solver for decay

problems or problems with small libraries. CRAM also allows for time dependent source terms and adjoint calculations, enabling the extension of ORIGEN to new classes of calculations. CRAM is currently available as an optional solver in stand-alone ORIGEN calculations.

ORIGAMI – New Advanced Used Fuel Assembly Characterization

SCALE 6.2 includes the new ORIGAMI (ORIGEN Assembly Isotopics) tool to compute detailed isotopic compositions for LWR assemblies containing UO_2 fuel by using the ORIGEN transmutation code with pre-generated ORIGEN libraries, for a specified assembly power distribution. The assembly may be represented by a single lumped model with only an axial power distribution, or by a square array of fuel pins with variable pin-powers as well as an axial distribution. In either case, ORIGAMI performs ORIGEN burnup calculations for each of the specified power regions to obtain the spatial distribution of isotopes in the burned fuel. Multiple cycles with varying burn-times and down-times may be used. ORIGAMI produces files containing SCALE and MCNP formatted composition data for axial burnup distribution at the last time-step, as well as a file containing the axial decay-heat at the final time-step, and a file specifying the energy-dependent radioactive source for use in shielding calculations.

ORIGEN Reactor Libraries

A new series of 1470 pre-generated burnup libraries for use in ORIGEN and ORIGAMI are introduced to SCALE 6.2. These libraries are generated under quality assurance with TRITON using the ENDF/B-VII.1 252-group nuclear data library through the use of a new template and script system. These libraries are intended to replace the ORIGEN-ARP libraries that were generated with a previous version of SCALE with varying levels of quality assurance. Libraries are available for a variety of fuel assemblies for commercial and research reactors, as detailed in Table 3 and Table 4.

Table 3. Assembly and Lattices Types with Available ORIGEN Reactor Libraries

Assembly type (Number of Libraries)		Lattice types
PWR	Babcock & Wilcox	15×15
	Westinghouse	14×14, 15×15, 17×17, 17×17-OFA
	Combustion Engineering	14×14, 16×16
	Siemens	14×14, 18×18
BWR	ABB	8×8-1
	Atrium	9×9-9, 10×10-9
	General Electric	8×8-4, 9×9-7, 7×7-0, 8×8-1, 8×8-2, 9×9-2, 10×10-8
	SVEA	64(8×8-1), 96(10×10-4), 100(10×10-0)
MOX	BWR Lattices (75)	ABB 8×8-1, Atrium 9×9-9, 10×10-9; GE 7×7-0, 8×8-1, 8×8-2, 9×9-2, 10×10-8; SVEA-64, 96, 100
	PWR Lattices (15)	Siemens 14×14, 18×18; CE 14×14, 16×16; B&W 15×15; Westinghouse 14×14, 15×15, 17×17, 17×17-OFA
Other	AGR (6)	
	CANDU (1)	19-pin, 28-pin, 37-pin
	IRT (12)	2M, 3M, 4M
	Magnox (4)	
	RBMK (24)	
	VVER-440 (3)	flat, radial enrichments (3.82, 4.25, 4.38)
	VVER-1000 (7)	flat enrichment

Table 4. Additional Information for ORIGEN Reactor Libraries

Assembly type	Enrichments [%]	Coolant/moderator densities [g/cm ³]	Maximum burnup [GWd/MTU]	Number of Libraries
PWR LEU	0.5, 1.5, 2, 3, 4, 5, 6	~0.73	70.5	7
BWR LEU	0.5, 1.5, 2, 3, 4, 5, 6	0.1, 0.3, 0.5, 0.7, 0.9	70.5	35
PWR MOX	*	~0.73	70.5	15
BWR MOX	*	0.1, 0.3, 0.5, 0.7, 0.9	70.5	75
AGR	0.5, 1.5, 2, 3, 4, 5	1.65	48.7	6
CANDU	0.711	0.8445	13.7	3
IRT	19.75, 36, 80, 90	0.989	159	12
Magnox	0.7, 0.8, 0.9, 1	1.628	13.7	4
RBMK	1.8, 2.2, 2.6, 3	0.15, 0.28, 0.41, 0.54, 0.67, 0.8	24.3	24
VVER-440	1.6, 2.4, 3.6, profiled	0.73	70.5	6
VVER-1000	0.5, 1.5, 2, 3, 4, 5, 6	0.7145	70.5	7

* Pu contents [%]: 4, 7, 10; ²³⁹Pu contents [%]: 50, 55, 60, 65, 70

ORIGAMI Automation Tool

ORIGAMI Automator, a graphical user interface integrated with Fulcrum facilitates the quantification of isotopics as a function of time for a large set of fuel assemblies. This tool was developed to support the Nuclear Regulatory Commission (NRC) in MELCOR analyses, but can be adapted to many other uses. The ORIGAMI Automator is an integrated graphical user interface that leverages many of modern components developed for SCALE 6.2 to enable the analyst to (1) easily create, view, and edit the reactor site and assembly information, (2) automatically create and run ORIGAMI inputs, and (3) analyze the results from ORIGAMI. The Automator uses the standard ORIGEN binary concentrations files produced by ORIGAMI, with concentrations available at all time points in each assembly's life. ORIGAMI Automator enables plotting of results such as mass, concentration, activity, and decay heat using an OPUS component. The ORIGAMI Automator Primer includes a description and user guide, a step-by-step tutorial for a simplified scenario, and appendices that document the file structures used.

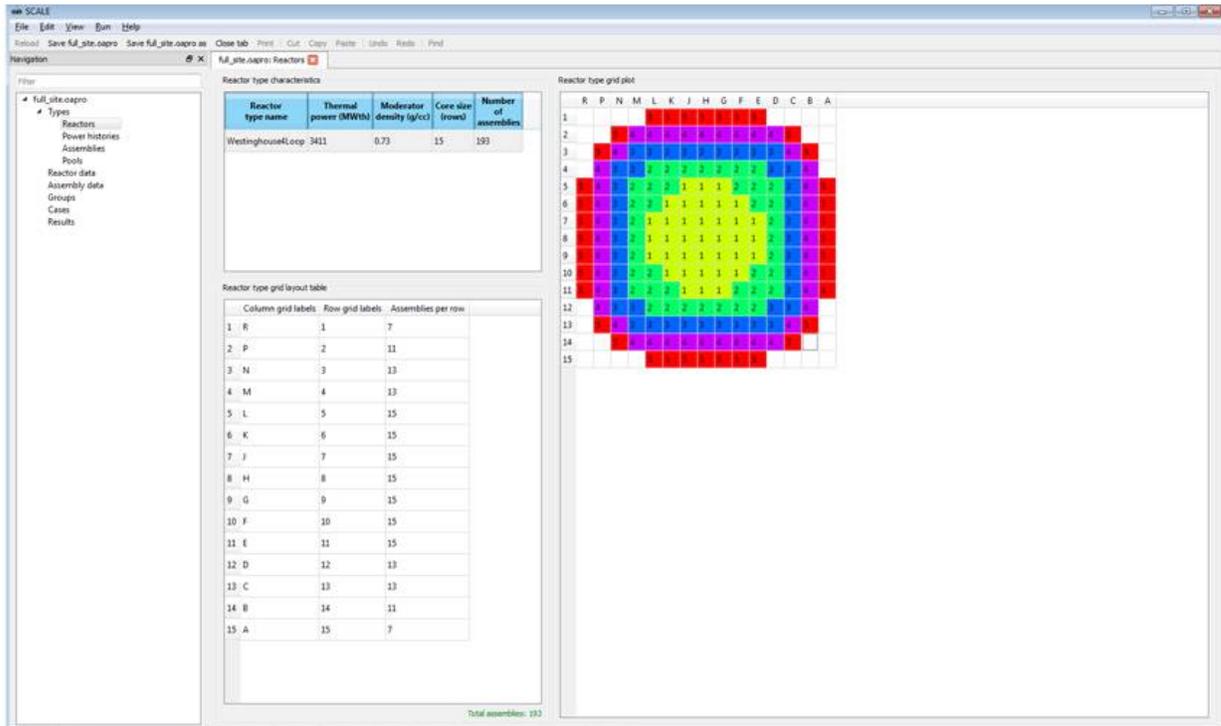


Figure 9. ORIGAMI Automator.

Radiation Shielding

The SCALE fixed-source Monte Carlo capability with automated variance reduction has been enhanced to enable CE calculations. The new CE capabilities will provide enhanced solution fidelity while still implementing the unique acceleration techniques of the FW-CADIS (Forward-Weighted Consistent Adjoint Driven Importance Sampling) methodology for deep penetration shielding and criticality accident alarm system modeling.

Continuous-energy Shielding Calculations

Prior to SCALE 6.2, the MAVRIC/Monaco capabilities relied on the MG approach for radiation transport. The MG approach is suitable for many applications but can be problematic for others. For example, deep penetration shielding through iron, where the MG-averaged cross section for capture resonances may not accurately represent the true transmission of neutrons due to fine “windows” through (i.e., minima in) the cross section. Additionally, MG cannot adequately resolve discrete gamma emission lines, such as those of ^{60}Co . With the CE treatment introduced in SCALE 6.2, improved solution fidelity is available. The generation of CE nuclear data and the implementation of CE physics in Monaco are based on a first-principles approach, where the simulation is represented as realistically as possible. This approach will lead to higher-fidelity results, but runtimes may be substantially increased over more approximate methods.

Monaco also includes capabilities to import sources generated with ORIGAMI and to access gamma sources directly from ORIGEN data.

Sensitivity and Uncertainty Analysis

CE Eigenvalue Sensitivity Analysis

With SCALE 6.2, the MG eigenvalue sensitivity and uncertainty analysis methods that use KENO for transport analysis are extended to provide CE capabilities through the implementation of the CLUTCH (Contributon-Linked eigenvalue sensitivity/Uncertainty estimation via Tracklength importance CHaracterization) methodology and IFP (Iterated Fission Probability) methods.¹² CLUTCH is an efficient methodology that has been demonstrated to provide high-fidelity results with manageable run-times and memory requirements. State-of-the-art sensitivity methods make CE calculations easier to use than MG in several ways; for example, CE calculations do not require resonance self-shielding calculations to determine implicit sensitivity effects, the simulation of a separate adjoint transport calculation, or the use of a flux mesh for tallying fluxes and flux moments.

The new capabilities enable improvements in sensitivity coefficient accuracy compared to MG. Table 5 compares the accuracy of the total nuclide sensitivity coefficients estimated by the each sensitivity method for the MIX-COMP-THERM-004-001 system.¹³ The MG analysis produced a ²³⁸U total nuclide sensitivity that disagreed with the reference direct perturbation sensitivity by 2.80 effective standard deviations (σ_{eff}), while the CE methods (IFP and CLUTCH) sensitivities were within almost 1 standard deviation of the reference sensitivity coefficients for all of the nuclides examined.

Figure 10 compares the figures of merit for the nuclide sensitivity coefficients presented in Table 5. Figure 10 indicates that the CE sensitivity calculations are of a comparable efficiency to the MG calculations, and in some cases the CE CLUTCH method is more efficient than MG method. This is remarkable because of the use of CE physics, which can impose relatively long cross-section lookup times, and indicates the efficiency of the CLUTCH method.

Table 5. MIX-COMP-THERM-004-001 total nuclide sensitivity coefficient comparison.

Sensitivity	Reference	MG TSUNAMI	CE TSUNAMI	
			IFP	CLUTCH
H ₂ O	0.2935 ± 0.0179	0.2805 ± 0.0088 (-0.65 σ_{eff})	0.2733 ± 0.0052 (-1.08 σ_{eff})	0.2793 ± 0.0065 (-0.75 σ_{eff})
²³⁸ U	-0.0061 ± 0.0003	-0.0050 ± 0.0002 (2.80 σ_{eff})	-0.0055 ± 0.0003 (1.16 σ_{eff})	-0.0057 ± 0.0001 (1.14 σ_{eff})
²³⁹ Pu	0.1262 ± 0.0087	0.1264 ± 0.0014 (0.02 σ_{eff})	0.1188 ± 0.0020 (-0.83 σ_{eff})	0.1190 ± 0.0001 (-0.83 σ_{eff})
²⁴⁰ Pu	-0.03777 ± 0.00350	-0.03750 ± 0.00011 (0.08 σ_{eff})	-0.03738 ± 0.00060 (0.11 σ_{eff})	-0.03743 ± 0.00002 (0.10 σ_{eff})
²⁴¹ Pu	0.00589 ± 0.00042	0.00599 ± 0.00004 (0.24 σ_{eff})	0.00567 ± 0.00014 (-0.50 σ_{eff})	0.00579 ± 0.00003 (-0.24 σ_{eff})

Figure 10. MIX-COMP-THERM-004-001 total nuclide sensitivity Figure of Merit comparison.

Table 6 gives the amount of computational memory required by each method for the sensitivity coefficient calculations. These memory requirements were obtained by subtracting the memory requirements of each eigenvalue-only calculation from the memory requirements of each sensitivity coefficient calculation. As seen below, the CLUTCH method excels in terms of memory usage, and can readily compute sensitivities for complex systems on a typical personal computer.

Table 6. Sensitivity method memory usage.

Model	MG TSUNAMI	CE TSUNAMI	
		IFP	CLUTCH
MIX-COMP-THERM-004-001	13,785 MB	10,643 MB	63 MB

TSUNAMI-3D also includes a Generalized Perturbation Theory (GPT) capability to compute the sensitivity of ratios of reaction rates to cross section data by introducing the GEneralized Adjoint Responses in Monte Carlo (GEAR-MC) method. The two GEAR-MC implementations are available for use in continuous-energy (CE) Monte Carlo simulations.

GPT sensitivity coefficients now share the same input at the previous multigroup GPT methods implemented in TSUNAMI-1D and TSUNAMI-2D. The Definitions block is used to define reaction rates in terms of material, reactions and energy ranges, and the SystemResponses block is used to define ratios for which sensitivity coefficients will be computed. A separate SDF file is generated each ratio defined in SystemResponses detailing the sensitivity of the reaction rate ratio to the cross-section data on an energy dependent basis. The uncertainty in each ratio due to cross-section covariance data is also provided in the output file.

Sampler

A new stochastic uncertainty quantification capability has been added with the SCALE 6.2 *Sampler* module that implements stochastic techniques to quantify the uncertainty in any computed result from any SCALE sequence due to uncertainties in

- neutron cross sections,
- fission yield and decay data, and
- any user input parameter, such as geometry, material density, isotopic composition, temperature, etc.

Sampler propagates these uncertainties through complex analysis sequences, such as depletion calculations, and provides the variation in the output quantities due to variations in any combination of input data, as shown in Figure 11.¹⁴ Correlations between systems with shared uncertainties are also computed, which is especially useful for quantifying correlated uncertainties in benchmark experiments, required for generalized linear least-squares techniques implemented by the SCALE module TSURFER.¹⁵

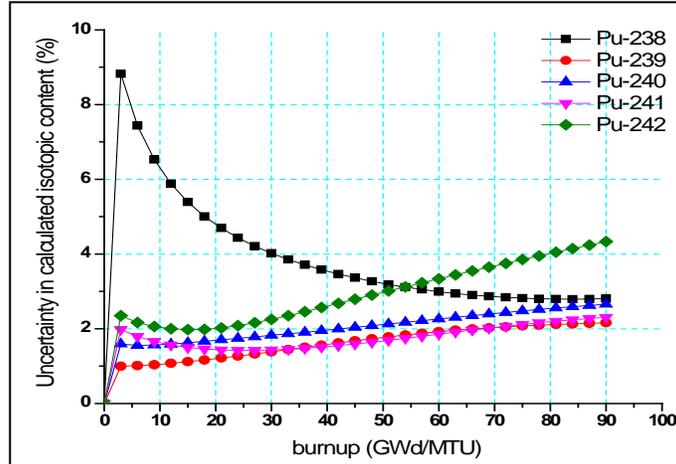


Figure 11. Uncertainty in plutonium isotopics as function of burnup.

Uncertainty libraries for fission product yields are implemented from ENDF/B-VII.0 data for the major actinides ^{235}U , ^{238}U , ^{239}Pu , and ^{241}Pu . These libraries are used directly by Sampler to assess uncertainty in fission product concentrations due to yield uncertainties. Covariance data are developed from fission models and a Bayesian method applied to ensure that uncertainties in the cumulative fission yields are consistent with the independent yields in each chain. The decay uncertainty data has 4 major components: decay constant uncertainty, q-value uncertainty, fraction of q value in photons, and branch ratio uncertainty. Perturbation factors for these 4 components are generated using uncertainties from the ENDF/B-VII.1 library. Correlations are accounted for decays with multiple decay modes to ensure the sum of branching is unity. No uncertainties are applied for data that does not have associated uncertainty information. Perturbations are applied using the covariance information with XSUSA to provide 1000 ORIGEN decay libraries. The uncertainty information has presently not been consolidated as a single file of perturbation factors.

Currently, decay data sampling is available for ORIGEN and ORIGAMI calculations, but is not operational with TRITON calculations without additional user-supplied preliminary calculations.

Note that Sampler can be built with MPI enabled for parallel computations (see Build Instructions below), but it can only sample the execution of serial codes. Threading is available with the default binary distribution (use the “-I” option of scalarte) when executing Sampler.

Other Updates

SCALE 6.2 includes a number of other improvements, such as the extension of the maximum allowed number of materials from around 2,000 to around 2 billion. Some older features are removed such as ENDF/B-V and -VI nuclear data, the point kernel shielding capability, and a material optimization search sequence. Additionally, the default temperature for materials with no temperature explicitly specified has been modified from 300K to 293K for improved consistency between MG and CE calculations.

Known Issues

In this release, there are a few issues scheduled to be resolved in the future:

1. *CELLMIX* and *DOUBLEHET* treatments are not implemented for radiation shielding problems using *MAVRIC*;
2. *CELLMIX* is not operational within *DOUBLHET* calculations to homogenize the entire pebble (not commonly used);
3. The *TRITON ASSIGN* capability is not implemented for cases that use *BRANCH*, *TIMETABLE*, or *SWAP* input blocks.

System Requirements

System Architecture:

- Linux 64-bit (RHEL 6)
- Mac OS X (Darwin) 10.9.5 or newer
- Windows 7, 64-bit or newer

System requirements:

Minimum requirements: 4 GB RAM per CPU, 40 GB of disk space + additional space to store output results

Recommended requirements: 8 GB RAM per CPU, 40 GB of disk space + 100 GB of scratch space + additional space to store output results

Production requirement for large models: 64 GB RAM, 40 GB of disk space + 500 GB of scratch space + additional space to store output results

Java Requirements:

Java1.6 or newer.

Installation Instructions

SCALE 6.2 installation has been broken into three components: 64-bit pre-compiled binaries, source code and the data. The pre-compiled binaries are available in platform familiar installers: NSIS on Windows, Drag-N-Drop bundle on Mac and Tar GZip on Linux. For deployments that include source code, it is available as a zip file and the data is available as a self-extracting IzPack installer. Other systems may require a custom build of SCALE, from source code, using the build instructions provided later in this document. Pre-Compiled Binaries

Note: **After you install the Binaries, you still must install the nuclear data.** After completing the binary installation following the instructions in this section, please follow the instructions for SCALE 6.2 Data.

Windows

After double-clicking the SCALE-6.2-setup.exe you may see a “loading...” dialog briefly popup.

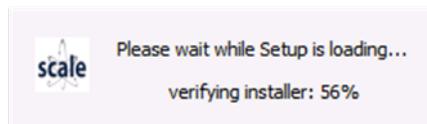


Figure 12. NSIS installer loading...

Once loaded, a “Welcome” dialog will prompt to begin the installation. Click “Next >” to continue.



Figure 13. NSIS “Welcome” dialog.

Next, the installer will prompt the “License Agreement”. Clicking “I Agree” will continue.

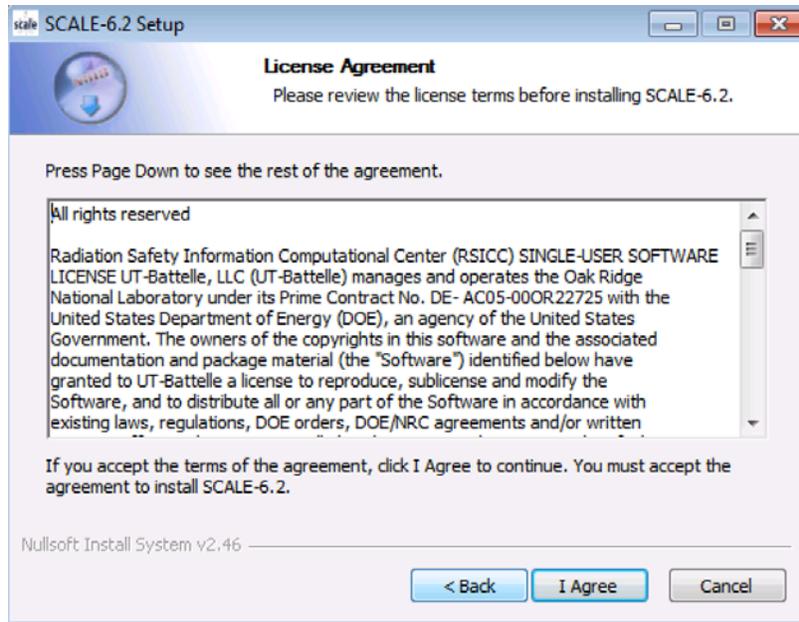


Figure 14. NSIS SCALE License Agreement.

Next, the “Choose Install Location” dialog is displayed. Clicking “Next >” will continue.

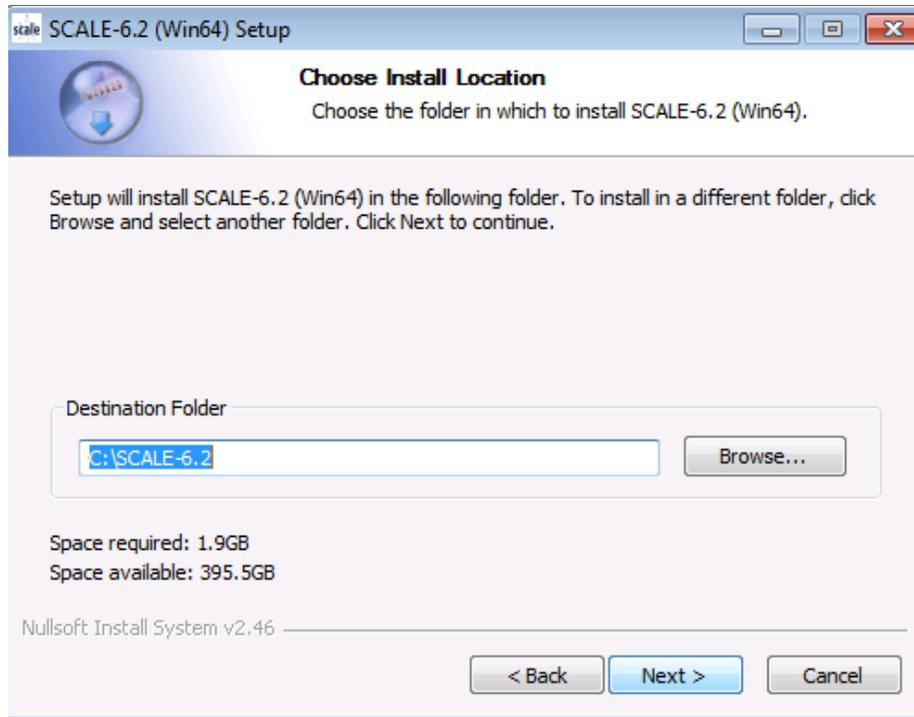


Figure 15. NSIS Choose Install Location.

Next, the “Choose Start Menu Folder” is prompted. This defaults to installing Start Menu shortcuts. You may click “Do not create shortcuts” to disable Start Menu shortcuts. Desktop shortcuts are always installed. Clicking “Install” will continue.

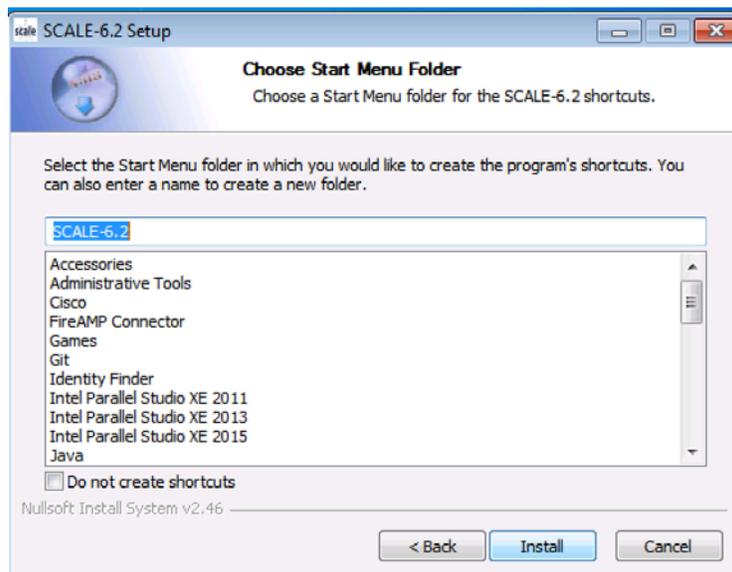


Figure 16. NSIS Choose Start Menu Folder.

Upon clicking “Install” the install starts and progress will be shown. Once complete, clicking “Next >” will continue.

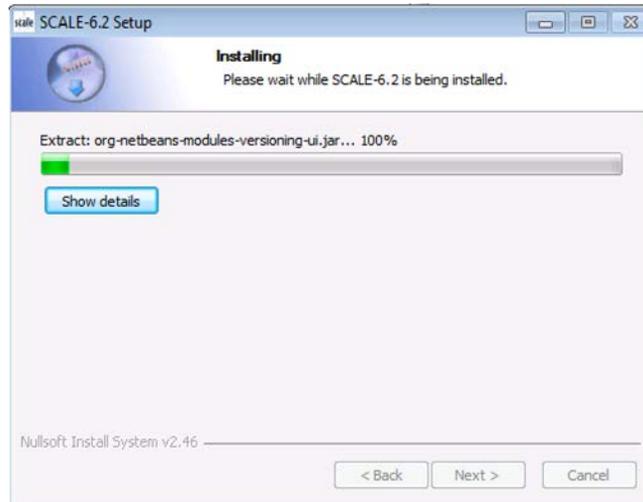


Figure 17. NSIS progress dialog.

Finally, installation is complete and you may click “Finish” to exit the NSIS installer, and then follow the instructions to install SCALE 6.2 Data.



Figure 18. NSIS complete dialog.

Note that Norton Antivirus on Windows requires an exception for the SCALE Runtime Environment program, scalerte.exe. If using Norton Antivirus, please establish this exception prior to installing or SCALE 6.2 will not be able to execute.

Mac OSX

After double-clicking the SCALE-6.2-setup.dmg you will see an opening dialog.

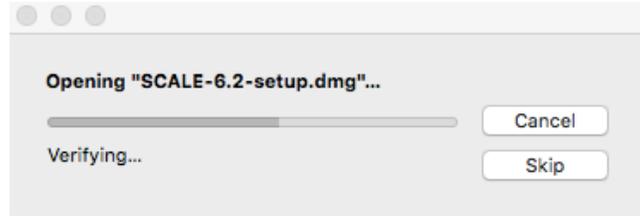


Figure 19. Mac OSX opening dialog.

Upon accepting, a Finder window will be raised showing “Applications” and “SCALE-6.2.app”. Simply drag “SCALE-6.2.app” onto “Applications” to install SCALE-6.2, and **then follow the instructions given below to install SCALE 6.2 Data.**

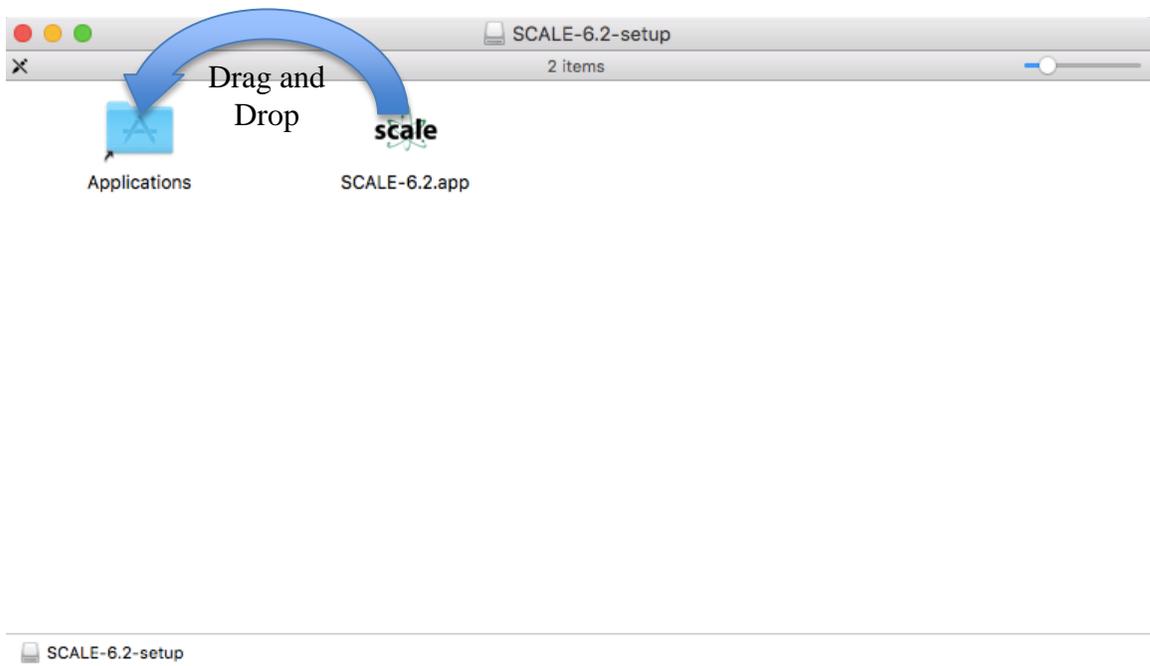


Figure 20. Mac OSX bundle contents.

Linux

Copy the SCALE-6.2.tar.gz to the installation directory. Invoke 'tar -xzf SCALE-6.2.tar.gz'. This will create the SCALE-6.2 directory. **Follow the instructions given below to install SCALE 6.2 Data.**

```
warroom3:~$ ls
SCALE-6.2.tar.gz
warroom3:~$ tar -xzf SCALE-6.2.tar.gz
warroom3:~$ ls
SCALE-6.2  SCALE-6.2.tar.gz
warroom3:~$ cd SCALE-6.2
warroom3:~$ ls
Ampx          Copyrights.txt  help          Kena3d        plugins       ScaleRunResources.xml
bin           etc             images       lib           python        us1stats
CentrmOptions.xml  exsite         include      MoreOptions.xml  regression    vbe
cmds          FileNameAliases.txt  Javapeno     ParmOptions.xml  samples
```

Figure 21. Linux Tar GZip.

Source Code

If you received the source code version, you can unzip SCALE-6.2-Source.zip file to any location on your computer.

For example, on a Windows platform, double click the SCALE-6.2.source.zip file.

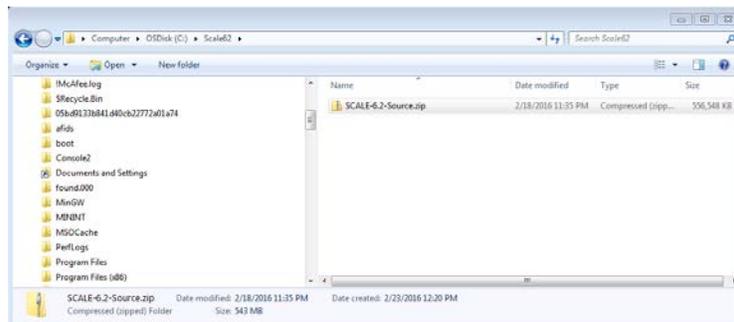


Figure 22. Windows Extract SCALE 6.2 source zip file.

This will extract the files into the “SCALE-6.2-serial-6.2-Source” directory, as shown below.

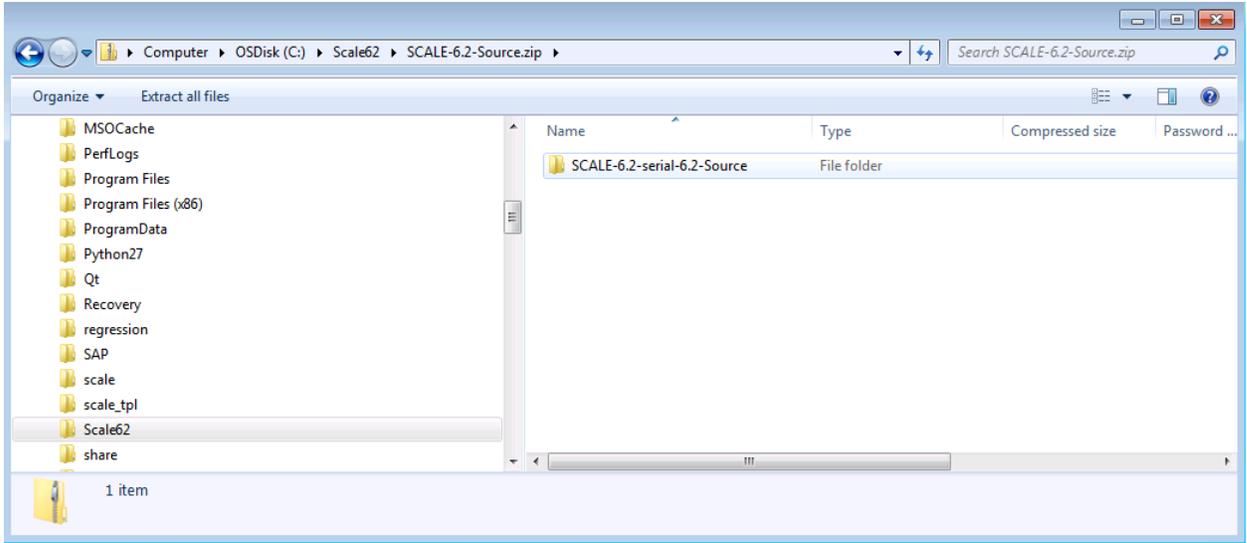


Figure 23. Windows SCALE 6.2 source directory.

The contents of the SCALE-6.2 source directory should be similar to this:

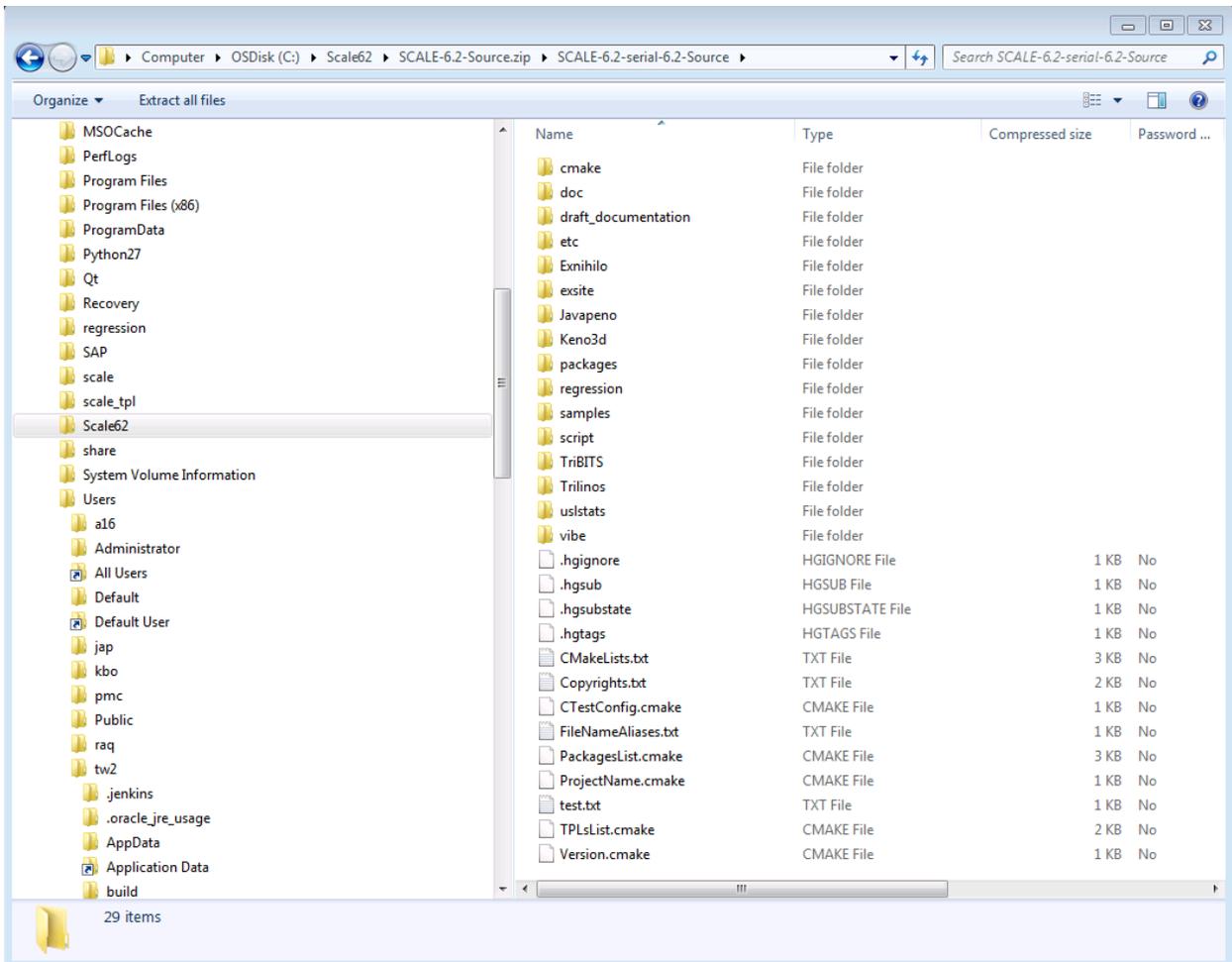


Figure 24. Windows Extract SCALE 6.2 source directory contents.

On Linux and Mac OSX create a destination directory for the source and copy the SCALE-6.2-Source.tar.gz into that directory.

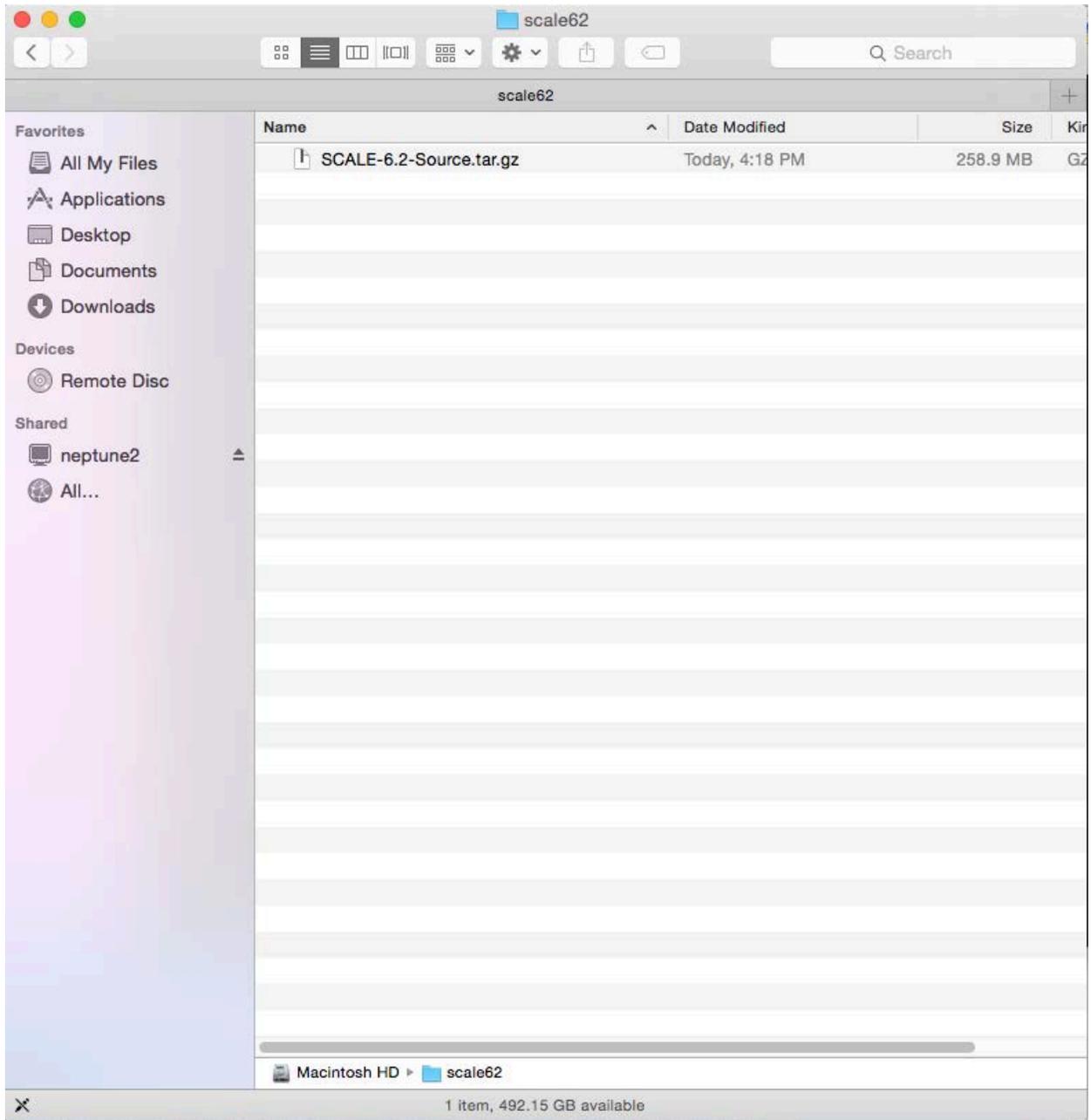


Figure 25. Mac SCALE 6.2 source directory prior to extracting source code.

Change to the directory and invoke `'tar -xzf SCALE-6.2-Source.tar.gz'` to deploy the source code.

SCALE 6.2 Data

To begin installation of SCALE 6.2 data, **copy the scale-6.2-data-setup.jar to your local disk. Double-click this jar file. If the installer does not start** then bring up a command prompt or terminal window and issue the following command: `java -jar scale-6.2-data-setup.jar` in the location where the installer jar file was copied.

After launching the installer, a dialog should appear as shown below.

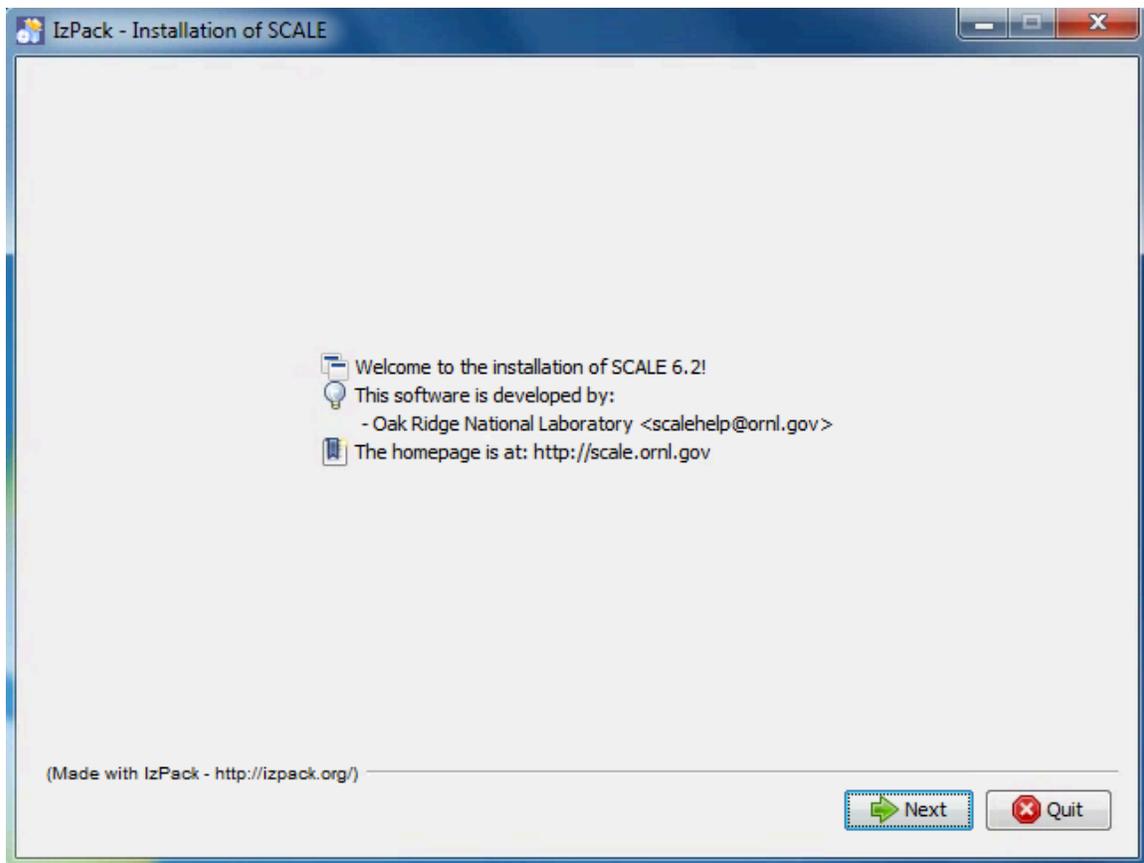


Figure 26. SCALE 6.2 data installer welcome dialog.

Please continue by pressing *Next*.

You will be prompted to review and accept the terms of the license agreement.

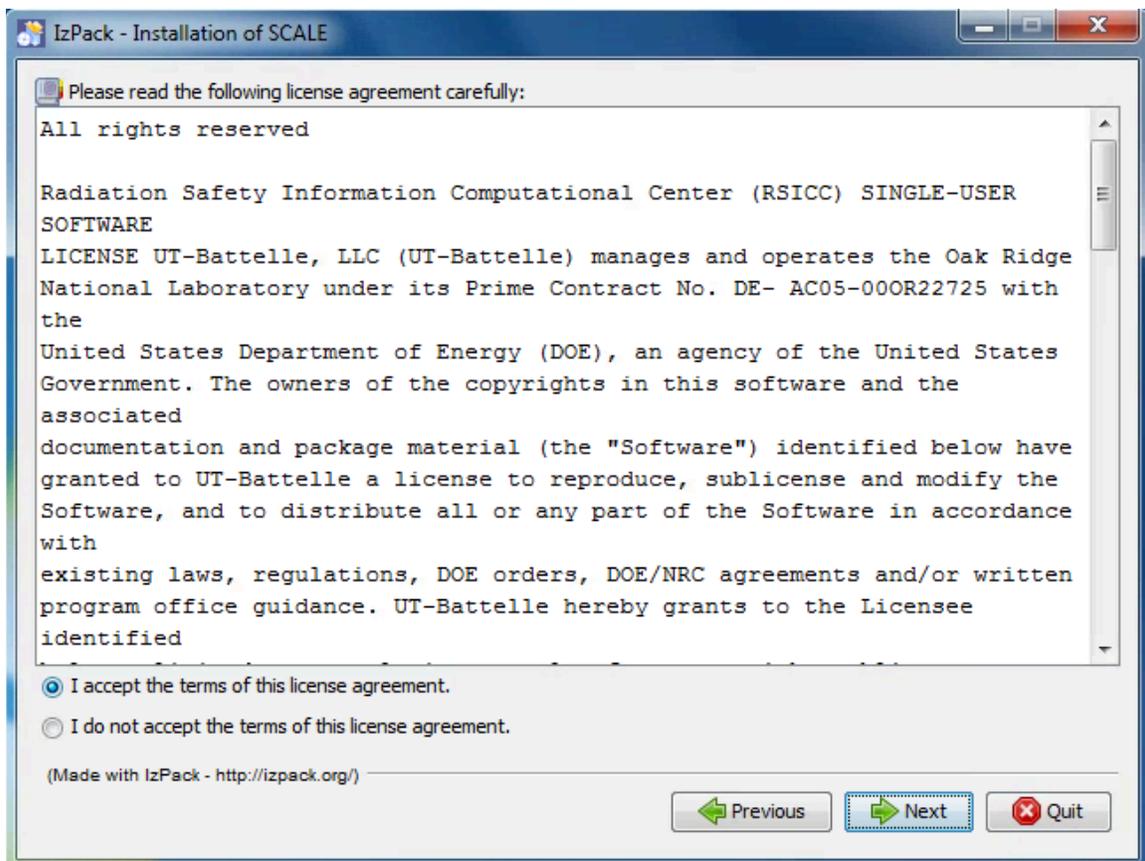


Figure 27. SCALE license agreement.

To proceed, check that you accept the terms and press next.

You will be prompted to choose the destination of your installation.

For Windows users, the recommended installation path is inside your SCALE-6.2 directory:

```
c:\SCALE-6.2
```

For Linux, a typical location is:

```
/scale/scale6.2
```

For Mac, a typical location is:

```
/Applications/SCALE-6.2.app/Contents/Resources
```

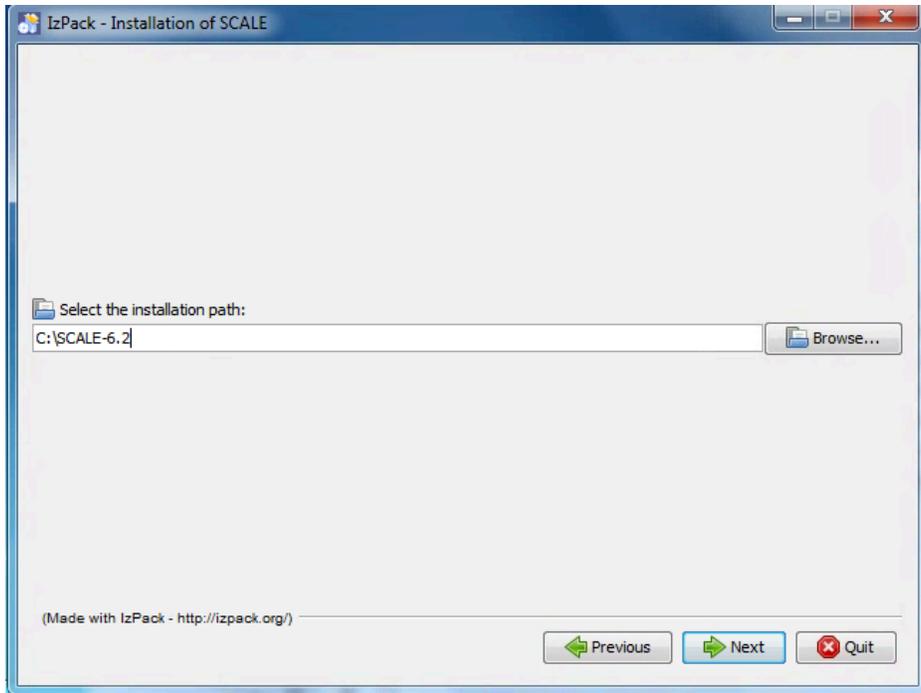


Figure 28. Setting typical data installation path on Windows.

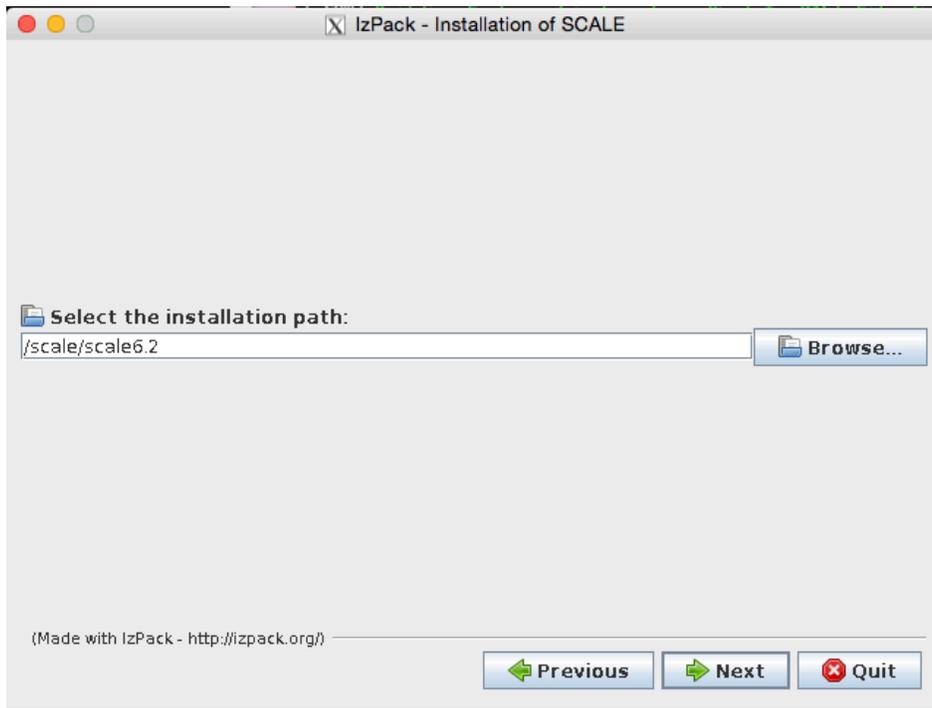


Figure 29. Setting a typical data installation path on Linux.

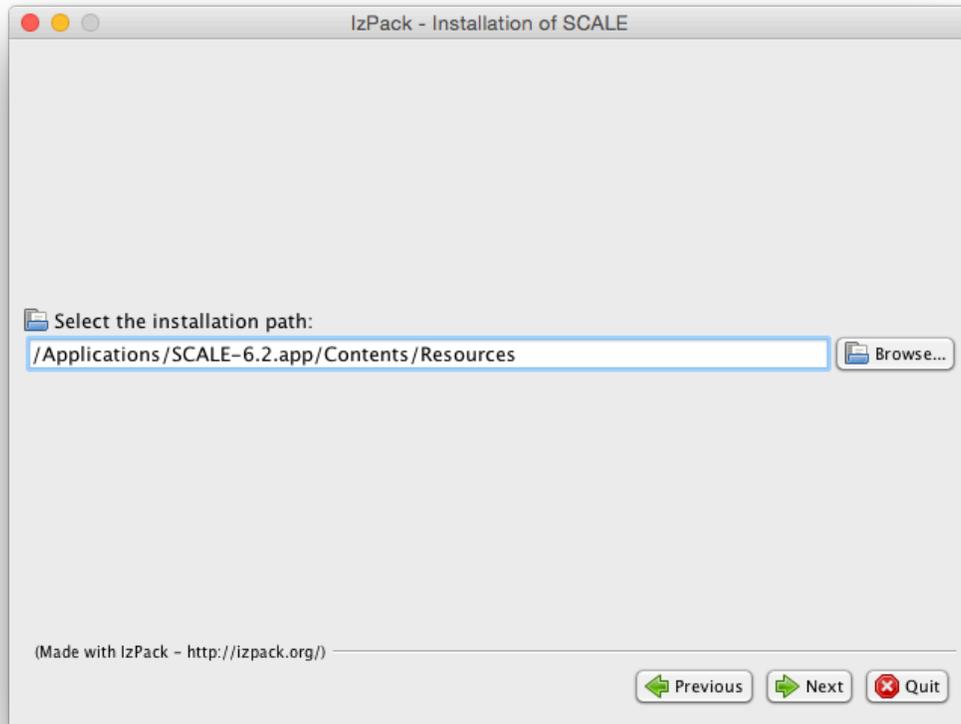


Figure 30. Setting a typical data installation path on Mac.

For some systems, the installer will not be allowed to create a new directory. If you encounter a message like the one shown below, simply create the directory manually (e.g. using Windows Explorer) then return to the installer and continue.



Figure 31. Data installer directory creation error message.

In most cases, the installer will present a prompt to confirm the creation of a new directory; if that target directory is correct you can simply press *OK*.

If the directory was manually created, the installer will notify you that it is about to overwrite any previous contents. Press *Yes* to continue.

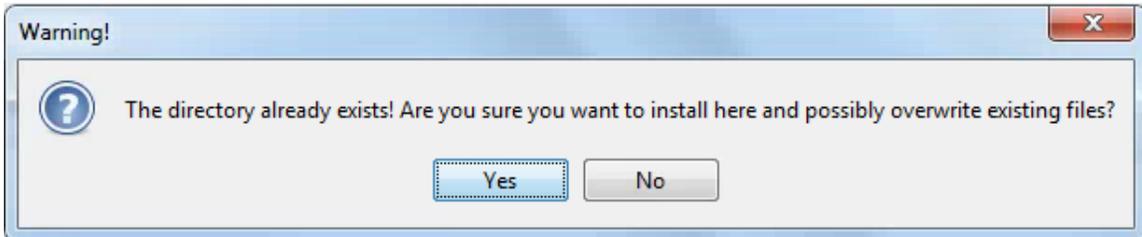


Figure 32. Directory overwrite dialog.

The installer then presents a dialog for available data sets you wish to install.

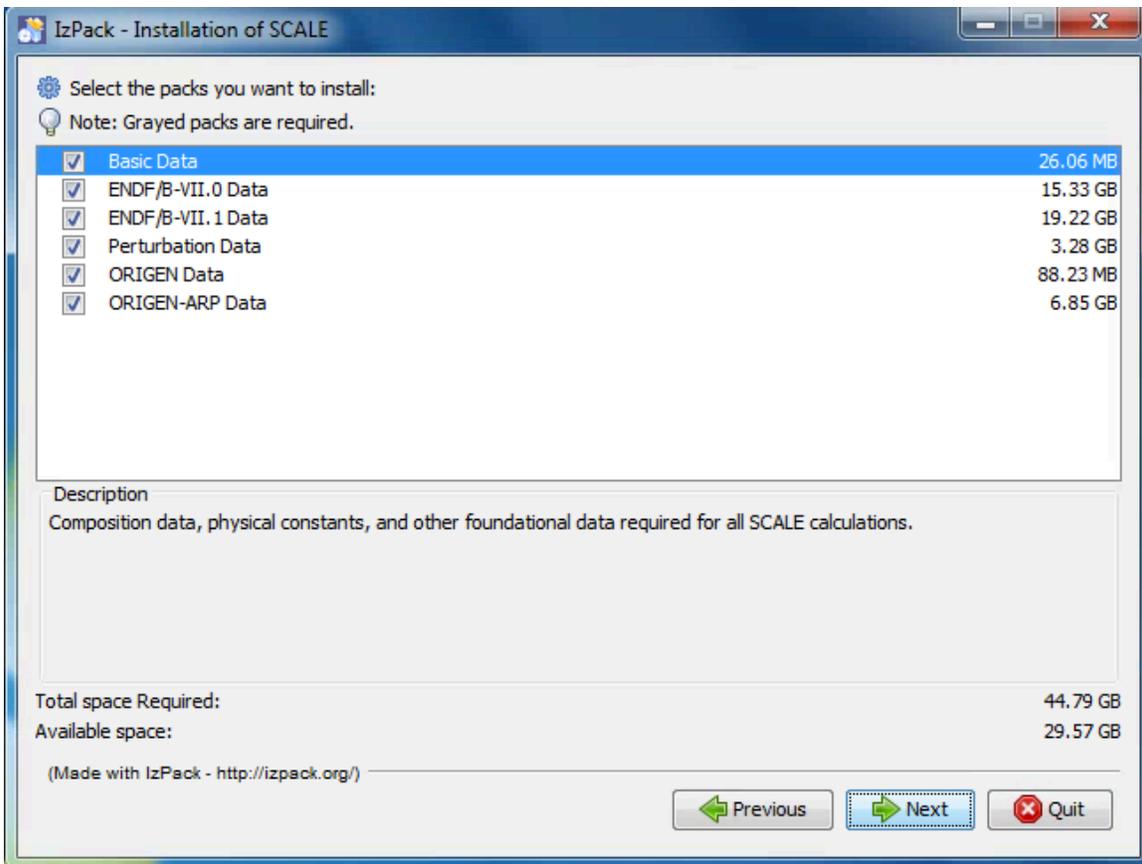


Figure 33. Data selection dialog.

Please select the data needed to perform your desired SCALE calculations.

- Basic Data — Composition data, physical constants, and other foundational data required for all SCALE calculations.
- ENDF/B-VII.0 Data — Nuclear Data from ENDF/B-VII.0 for continuous-energy and multi-group calculations; Coupled n-gamma data from ENDF/B-VII.0 for multi-group calculations; SCALE 6.0/6.1 44-group neutron covariance data. Optional for most calculations, but provided for backwards compatibility.
- ENDF/B-VII.1 Data — Nuclear Data from ENDF/B-VII.1 for continuous-energy and multi-group calculations; Coupled n-gamma data from ENDF/B-VII.1 for multi-group calculations; ENDF/B-VII.1 neutron covariance data. Recommended for most calculations and required to run the SCALE sample problems.
- Perturbation data — 1000 samples of data each for: ENDF/B-VII.1 neutron covariance data, SCALE 6.0/6.1 neutron covariance data, fission product yield covariance data, and radioactive decay covariance data. Required for Sampler calculations.
- ORIGEN Data — Activation, depletion and decay data from ENDF/V-VII.1 and JEFF 3.1. Required for activation, depletion and decay calculations.
- ORIGEN-ARP Data — Pre-generated ORIGEN reactor libraries for many fuel types. Required for spent fuel characterization and source terms calculations.

Next, the installer may ask for the location of the data *.pak* files that are unpacked during the installation process. Please direct the installer to the location of these data files, which are part of the SCALE distribution and may be on the distribution media or copies to a local directory as shown below.

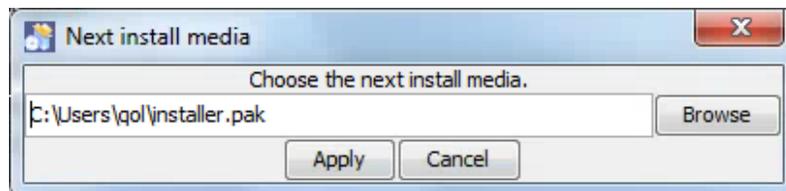


Figure 34. Media selection dialog.

The installer will post the progress of the installation. If installing from a DVD set, part way through the installation you will be prompted for *installer.pak.1*. Please insert the PAK.1 disk and click *Apply*.

The installation from PAK.1 will complete and prompt for *installer.pak.2*. Please insert the PAK.2 disk and click *Apply*.

PAK.2 will finish. Proceed with all DVDs or files in the delivery until you have completed your installation of SCALE 6.2.

Running SCALE

SCALE is run by using the Fulcrum user interface or by invoking the SCALE Runtime Environment, *scalerte*, from the command line. Note that this release does not include the previous interface GeeWiz or OrigenArp for running SCALE on Windows, as Fulcrum should be used instead. Additionally, previous shortcuts like *runscale* are not implemented.

Running SCALE from Fulcrum

The most convenient way to run SCALE from a desktop is by launching Fulcrum. The Fulcrum executable is provided in the *bin* directory where SCALE was installed (e.g. *C:\SCALE-6.2\bin\Fulcrum.exe*). Fulcrum includes an online help document to assist users with its many features and includes links to the user manual and primers.

Windows

For a Windows installation, a Fulcrum shortcut is provided in the *SCALE-6.2* shortcuts folder on the desktop.

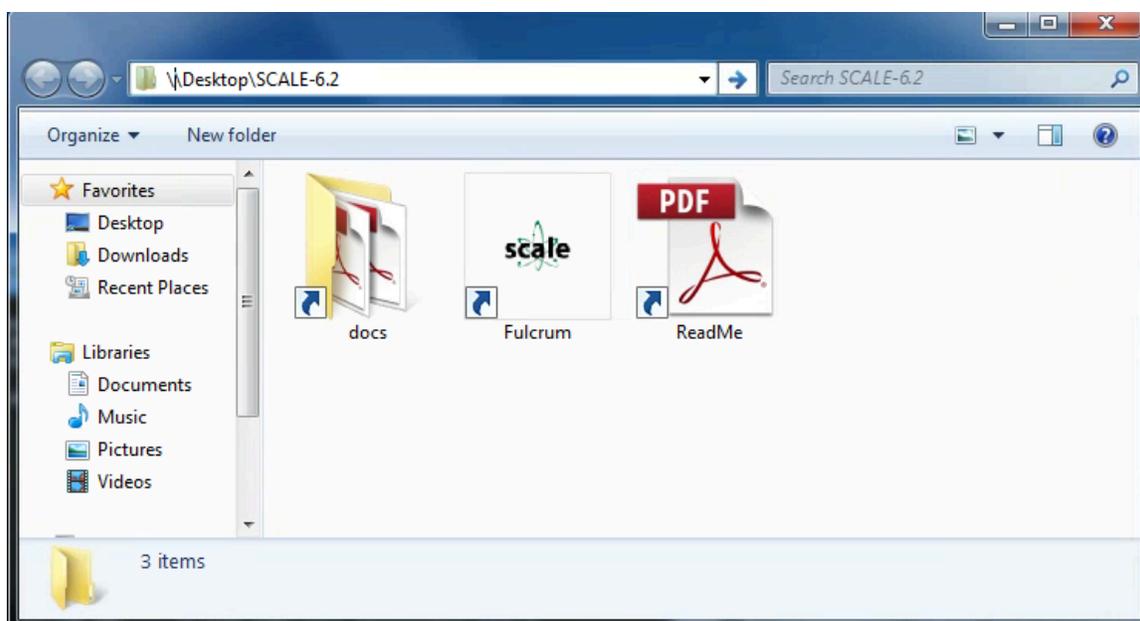


Figure 35. Windows desktop SCALE-6.2 shortcuts folder.

Linux

For Linux, launch Fulcrum from the bin directory of the SCALE installation (e.g. */scale/scale6.2/bin/Fulcrum*).

Mac

For a Mac installation, please use the *SCALE-6.2* app in the Applications directory.

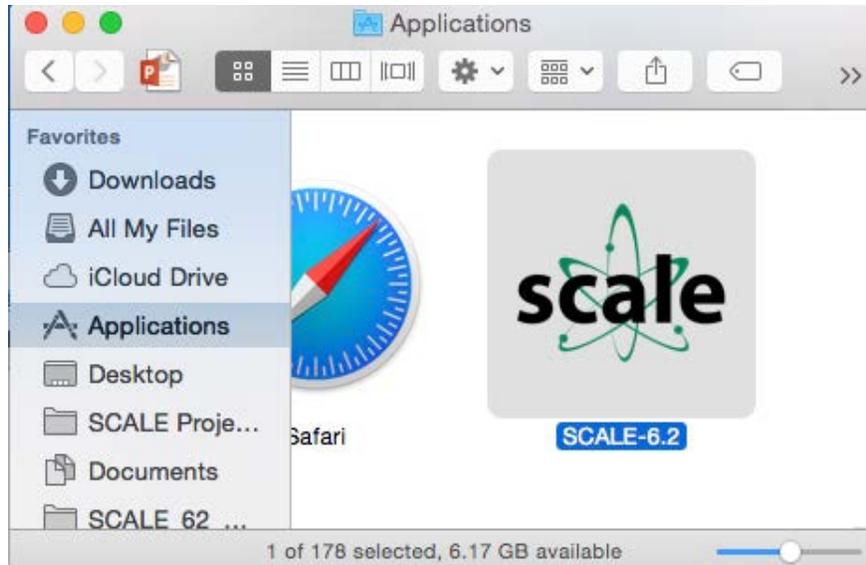


Figure 36. Mac SCALE-6.2 app in the Applications folder.

Running SCALE from the Command Line

Using the command line, SCALE can be executed using the *scalerte* command from the *bin* directory inside the SCALE installation. Paths for each platform are shown below based on recommended installation directories. Your directory may differ based on the installation.

Windows

```
C:\SCALE -6.2\bin\scalerte.exe
```

*Note that Norton Antivirus may require an exception for *scalerte.exe* on Windows to prevent the executable from being quarantined.

Linux

```
/scale/scale6.2/bin/scalerte
```

Mac

```
/Applications/SCALE -6.2.app/Contents/Resources/bin/scalerte
```

The SCALE runtime environment provides several command line options. The usage is as follows.

```
scalerte [options] inputfile(s) [options] [inputfile(s)]
```

Where *inputfile(s)* are one or more files or file patterns (*test.inp*, or *test*.inp*, etc.).

Where *options* are:

-a: Specify alias file.

```
-a path/to/aliasesfile
```

-f: Add hostname to output filename. Produces inputfile.hostname.out

-h: Print this information as a help message

-I: Number of threads to use for MPI/OpenMP directives. -I 4

-m: Print information messages as SCALE executes

-M: Specify a machine names file for SCALE parallel capabilities.

```
-M /path/to/machine/names/file
```

-n: Nice level on Nix systems, ignored on Windows. Default: -n 2

-N: Number of MPI processes to run. -N 20

-o: Overrides the default *inputfile.out* output name. The *.out* extension is appended by scalerte, so there is no need to specify the extension.

```
-o path/to/outputfile
```

NOTE: If the *path/to/outputfile* already exists, it will be deleted. If this option is specified while in stack mode (multiple input files), the value provided is prepended to the inputfile's basename.

```
scalerte triton* -o myout results in output names  
myout.triton*.out
```

-r: Keep the temporary working directory after execution

-t: No new temporary working directory. Uses last temporary working directory, -r is implied.

-T: Specify temporary working directory.

```
-T directory/path
```

NOTE: If -T is specified while in stack mode (multiple input files) the value provided is appended with the index of the file.

```
scalerte t0.inp t1.inp t2.inp -T mytmp
```

results in *mytmp*, *mytmp1*, and *mytmp2* temporary directories.

-v: Turn on verbose activity printing for scalerte.

-V: Print the scalerte version date.

- x: Do not return XSDRNPM output in a *.xsdrnfiles* directory.
- z: Add date to the output filename.

Produces output files in the form of
inputfile.yyyy.MM.ddThh.mm.ss.out

Where:

- *yyyy*- is the year of execution.
- *MM*- is the month of execution.
- *dd*- is the day of execution.
- *hh*- is the hour of execution.
- *mm*- is the minute of execution.
- *ss*- is the second of execution.

Example Invocation

For users familiar with previous invocation of the SCALE batch script, this usage is no longer valid. The only valid entry point is *scalerte*. *Scalerte* can be executed from anywhere with a fully qualified path such as *c:\SCALE-6.2\bin\scalerte* or */scale/scale6.2/bin/scalerte*, for example.

Invoke SCALE on a single input file named *HelloWorld.inp*

```
scalerte HelloWorld
```

or

```
scalerte HelloWorld.inp
```

Invoke SCALE on all input files patterned *HelloWorld*.inp*

```
scalerte HelloWorld*.inp
```

Invoke SCALE on all input files patterned *HelloWorld*.inp* and print runtime messages to the console

```
scalerte HelloWorld*.inp -m
```

Invoke SCALE on all input files patterned *HelloWorld*.inp* and include hostname and date/time in the output file's name

```
scalerte HelloWorld*.inp -fz
```

or

```
scalerte HelloWorld*.inp -f -z
```

Invoke SCALE on *HelloWorld.inp* and rename output to be *MyHello.out*

```
scalerte HelloWorld -o MyHello
```

or

```
scalerte HelloWorld.inp -o MyHello
```

Invoke SCALE on all files patterned *HelloWorld*.inp* and rename output to be *MyHelloWorld*.out*.

NOTE: When SCALE is run in stack mode (multiple inputs), the output override is prepended to the input file's name.

```
scalerte HelloWorld*.inp -o My
```

Invoke SCALE on *HelloWorld.inp* and keep the working directory.

```
scalerte HelloWorld.inp -r
```

Invoke SCALE on *HelloWorld.inp* and override and keep the working directory.

```
scalerte HelloWorld.inp -r -T myHelloWorldTempDir
```

Invoke SCALE on *HelloWorld.inp* and specify the number of threads to be 4.

```
scalerte HelloWorld.inp -I 4
```

SCALE Variables

This section describes the environment variable used within *scalerte*. These variables can be accessed through SCALE's *shell* module to populate the working directory, and/or to return SCALE-generated files that are not returned by *scalerte*.

Shell is used to perform system commands inside a problem either before, after, or between explicitly called modules. It is usually used in SCALE to link a file from one name and place to another name and place, delete files or directories, and move or copy files. The use of shell in an input has the following form:

```
=shell
```

```
*** System Commands ***
```

```
end
```

Where *System Commands* are any UNIX or DOS command(s). This is particularly useful in tandem with the following environment variables when files not automatically returned are desired. The following example shell command copies the x16 file back alongside the output file.

```
=shell
```

```
cp x16 ${OUTBASENAME}.x16
```

end

Please note the syntax of Nix and Windows environment variables, `${VAR}` and `%VAR%` respectively, are interchangeable as *scalerte* and shell understand both.

There are seven primary locations known by *scalerte*. These primary locations are:

- The user's home directory, HOME.

*Nix systems, `${HOME}`, `/home/uid`.

*Windows, `%HOME%`, `C:\Users\uid`.

- The directory of SCALE, SCALE.

*Nix systems, `${SCALE}`, location of user's installation. Typically `/scale/scale#`.

*Windows, `%SCALE%`, location of user's installation. Typically `C:\SCALE#`.

- The directory of the input file, INPDIR.

*Nix systems, `${INPDIR}`.

*Windows, `%INPDIR%`.

- The directory of the output file, OUTDIR, by default, this is the same as INPDIR, because the output file is written next to the input file.

*Nix systems, `${OUTDIR}`.

*Windows, `%OUTDIR%`.

- The directory from which SCALE was invoked, the return directory, RTNDIR. This is the directory your console will return to upon completion.

*Nix systems, `${RTNDIR}`.

*Windows, `%RTNDIR%`.

- The directory that contains the SCALE data, DATA.

*Nix systems, `${DATA}`.

*Windows, `%DATA%`.

- The working directory for a given input file, TMPDIR, or shorthand TMP.

*Nix systems, `${TMPDIR}`, `${TMP}`.

*Windows, `%TMPDIR%`, `%TMP%`.

There are several secondary locations, located in the SCALE directory tree. These are as follows:

- The directory containing the platform-specific compiled programs, PGMDIR, or legacy PGM_DIR.

 - *Nix systems, `${PGMDIR}`, `${PGM_DIR}`.

 - *Windows, `%PGMDIR%`, `%PGM_DIR%`.

Lastly, there are several environment variables provided for convenience, and/or associated with output data that can be useful.

- The directory containing the ORIGEN data files, ORIGENDIR.

 - *Nix systems, `${ORIGENDIR}`.

 - *Windows, `%ORIGENDIR%`.

- The base name of the input file, BASENAME. This is the name of the input file without both absolute path and extension.

 - *Nix systems, `${BASENAME}`, or `${CASE_NAME}`.

 - *Windows, `%BASENAME%`, or `%CASE_NAME%`.

- The base name of the output file, OUTBASENAME, or legacy CASE_NAME. This is the name of the output file without both absolute path and extension.

 - *Nix systems, `${OUTBASENAME}`, or `${CASE_NAME}`.

 - *Windows, `%OUTBASENAME%`, or `%CASE_NAME%`.

- The base name of the output file, OUTBASE. This is the absolute name of the output file without absolute file extension.

 - *Nix systems, `${OUTBASE}`.

 - *Windows, `%OUTBASE%`.

- The absolute path to the input file, INPUTFILE.

 - *Nix systems, `${INPUTFILE}`.

 - *Windows, `%INPUTFILE%`.

- The absolute path to the output file, OUTFILE.

 - *Nix systems, `${OUTFILE}`.

 - *Windows, `%OUTFILE%`.

- The directory containing USLSTATS output, USLDIR. If USLSTATS data were output, these data will be located in OUTDIR\OUTBASENAME.uslstats directory.

*Nix systems, \${USLDIR}.

*Windows, %USLDIR%.

- The directory containing CENTRM output, CENTRMDIR. If CENTRM data were output, these data will be located in OUTDIR\OUTBASENAME.centrmfiles directory.

*Nix systems, \$CENTRMDIR.

*Windows, %CENTRMDIR%.

- The directory containing XSDRNPM output, XSDRNDIR. If XSDRNPM data were output, these data will be located in OUTDIR\OUTBASENAME.xsdrnfiles directory.

*Nix systems, \$XSDRNDIR.

*Windows, %XSDRNDIR%.

- The platform specific file separator, FS. This is either back slash (\) on Windows, or forward slash (/) on Nix systems.

*Nix systems, \${FS}.

*Windows, %FS%.

Parallel Execution Capability

SCALE 6.2 contains five modules and sequences that have distributed memory (MPI) parallelism; KENO V.a, KENO-VI, Sampler, ORIGAMI, and Runner. However, the binary executable files distributed with SCALE do not have MPI enabled. In order to run one of these codes in parallel the user must first build SCALE with MPI enabled (see build instructions in this guide). Control modules, like CSAS6, T6-DEPL, and TSUNAMI-3D-K6, automatically initiate the parallel version of KENO-VI in a parallel SCALE build if the user provides the required arguments as summarized below. When running a standalone code in parallel (such as KENO-VI), a “%” prefix on the sequence specification record in the input file (e.g. =%keno-vi). Parallel code execution is currently available on Linux and Mac systems but is not available on Windows PCs.

The executable binary code distributed with SCALE only enables serial calculations. If MPI parallelism is desired, the source code must be compiled with MPI support enabled for the particular platform and configuration where the code will be executed in parallel. The SCALE build configuration sets some variables (*SCALEMPI*, *SCALECMDS*, and *MPIRUN*) depending on the third party MPI package for the SCALE driver, which prepares two different execution environments with these variable sets for both serial and parallel code execution.

Executing SCALE in parallel is initiated by *scalerte* when the user provides the necessary MPI command line arguments with the two options, *-N* and *-M*, which specify the number of MPI processes and machine names, respectively. To run an input called *HelloWorld.inp* from an MPI build on SCALE on two nodes specified in the machine file *mach* use the command:

```
scalerte -N 2 -M mach HelloWorld.inp
```

With these options, *scalerte* sets *NTASKS* and *MACHINEFILE* variables depending on the user request, and passes them to the SCALE driver. During the parallel code execution process, the driver invokes the MPIRUN wrapper and passes some variables¹ to all MPI processes to setup run time environment on each process and to run the parallel functional module across MPI processes. An additional variable *MPIARGS* was also defined in the driver to pass some user defined MPI options to MPIRUN wrapper.

¹ By default, the driver passes the following environment variables to all MPI nodes: *PATH*, *SCALE*, *RTNDIR*, *INPDIR*, *PGMDIR*, *TMPDIR*, *DATA*, *NTHREADS*, *SYSTMP*, *PROCESSID*.

SCALE Sample Problems

The SCALE sample problems are carefully designed by the SCALE developers to verify the installation and functionality of SCALE relative to expected results. Users are urged to run the sample problems to verify the proper installation of SCALE. ORNL has provided a set of reference results from each sample problem against which the results of each installation can be compared.

The Sample Problems are most easily run through Fulcrum by selecting *Run SCALE Verification* from the *Run* menu on the main menubar. This will run all of the Sample Problems sequentially and present any differences from the ORNL-generated results.

The SCALE Runtime Environment, *scalerte* described in the subsequent section, has a built-in scripting interface to allow for invocation of several groupings of sample problems, from individual problems, problems for a specific module, subsets of modules, or all samples. The sample problems will print a message indicating the currently running sample problem, followed by the differences between ORNL-generated results and the newly generated results for the sample problem(s). Note that there may be a significant delay (i.e., as much as an hour) before messages are printed to the console. The final message will indicate that the “Process finished with a 0 return code; . . .” If no differences or minimal differences are noted for a particular sample problem, SCALE has been properly installed for the tested functionality.

The set of sample problems is located in the samples directory of the SCALE installation.

To run all sample problems, execute the following command.

```
scalerte @samples/samples
```

To invoke sample problems for a particular module or sequence, use

```
scalerte @samples/samples modulename
```

where *modulename* is the name of the module or sequence to test, e.g. *centrm*

To invoke a single sample problem, do the following.

```
scalerte @samples/samples problemname
```

where *problemname* is the name of the specific sample problem to test, e.g., *centrm-pwr.inp*

The above *modulename* and *problemname* examples can be combined and repeated to exercise sets of sample problems of interest. E.g., run *csas* and *origen* sequence sample problems in addition to the *centrm-pwr.inp* sample.

```
scalerte @samples/samples csas centrm-pwr.inp origen
```

As the sample problems execute, feedback will be provided to the screen such as shown in the example below. Where no differences or only small differences are reported

between the ORNL results and the currently generated results for each sample, SCALE has been properly installed and configured for the tested functionality.

```
Command Prompt - c:\scale6.2\bin\scalerte @samples\samplesampler
Now running the neut3 sample problem at Mon 15/07/2013 22:24:03...
The neut3 sample problem has finished at Mon 15/07/2013 22:30:25.
files are identical.
-----
Now running the neut2 sample problem at Mon 15/07/2013 22:30:25...
The neut2 sample problem has finished at Mon 15/07/2013 22:32:51.
files are identical.
-----
Now running the neut1 sample problem at Mon 15/07/2013 22:32:51...
The neut1 sample problem has finished at Mon 15/07/2013 22:33:29.
files are identical.
-----
Now running the mcdancoff sample problem at Mon 15/07/2013 22:33:29...
The mcdancoff sample problem has finished at Mon 15/07/2013 22:34:14.
files are identical.
-----
Now running the mauricUtilities4 sample problem at Mon 15/07/2013 22:34:15...
The mauricUtilities4 sample problem has finished at Mon 15/07/2013 22:34:26.
files are identical.
-----
Now running the mauricUtilities3 sample problem at Mon 15/07/2013 22:34:26...
The mauricUtilities3 sample problem has finished at Mon 15/07/2013 22:36:01.
files are identical.
-----
```

Figure 37. Sample problem output messages.

For any problems or questions, please contact scalehelp@ornl.gov.

Build Instructions

These instructions are only for those who wish to recompile their SCALE binaries. If you are running SCALE using the precompiled binary executable files distributed with SCALE, please disregard this section.

For SCALE 6.2, the build configuration has been completely renewed relative to all previous versions. The SCALE build is now based on CMake from KitWare, which supports a consistent experience on Linux, Mac, and Windows.

Overview

There are four main steps to create and install SCALE binaries

1. Install compilers and third-party libraries (TPLs)
2. CMake configuration - This generates a native build tree
3. Compilation - This compiles all executables and libraries
4. Installation - This deploys all executables into a configuration ready for execution

Required Resources

SCALE required the following programs in order to compile:

1. Fortran Compiler
 - i. Windows – Intel Fortran compiler 15.0.
 - ii. Linux - Intel Fortran compiler 14.1+/GNU gfortran 4.8.3+ compiler.
 - iii. Mac OS – GNU gfortran 4.8.5+ compiler.
2. C/C++ Compiler
 - i. Windows – Intel C/C++ compiler 15.0.
 - ii. Linux – Intel C/C++ compiler 14.1+ / GNU g++/gcc 4.8.3+ compiler.
 - iii. Mac OS – GNU g++/gcc 4.8.5+ compiler.
3. CMake 2.8.12.2+ - Platform independent build configuration.

SCALE requires the following libraries, which are not distributed with SCALE:

1. LAPACK libraries:module
 - a. lapack
 - b. blas

You must modify the LAPACK variable in `configure_scale_gnu.sh` to point to your installation of LAPACK prior to configuration.

Following optional package is required by SCALE if MPI support is desired:

1. OpenMPI-1.8.1+ - Optional for SCALE build with MPI support on Linux and Mac OS (currently, SCALE does not support MPI for Windows)

Mac OSX Resources

Mac OSX does not come preloaded with the necessary resources for compiling SCALE. Macports provides a simple mechanism for retrieving almost all of the required resources necessary to compile SCALE on a Mac. This process has only been tested on MacOSX 10.9.5.

You can download the latest macports from: <http://www.macports.org/>. You will need XCode(Free from App Store), with commandline tools installed, for macports to work.

You will need administrative privileges to install macports and the necessary resources.

Please see macports instructions for xcode installation/requirements.
<http://guide.macports.org/#installing.xcode>

After installing these resources, a new shell session is necessary to access the new tools.

NOTE if your rsync port is blocked by for firewall (as is the case at ORNL), you may synchronize over http by updating your */opt/local/etc/macports/sources.conf* file to change the line:

```
rsync://rsync.macports.org/release/tarballs/ports.tar [default]
```

to

```
http://www.macports.org/files/ports.tar.gz [default]
```

The following commands will install all necessary resources.

If rsync port is blocked use:

```
$> sudo port -d sync
```

If rsync port is open:

```
$>sudo port selfupdate
```

Regardless of rsync status execute the following commands:

```
$>sudo port install gcc48
```

```
$>sudo port select gcc mp-gcc48
```

```
$>sudo port install qt4-mac
```

These commands upgrade the default compilers from gcc-4.2.2 to gcc-4.8.3 as well as install qt4.8.6.

CMake Configuration

Throughout SCALE you will find 'CMakeLists.txt' files. From the SCALE root directory these CMakeLists.txt files create a tree of included directories called the SOURCE TREE. Namely the source directories:

```
packages/AmpxLib
    XSProc
    ScaleLib
    etc..
```

To configure a build call 'cmake' on the root CMakeLists.txt file, namely `scale_dir/CMakeLists.txt`. CMake takes your source tree and creates a BUILD TREE. The build tree contains or will contain your build configuration, Make or NMake files, and all compilation output, i.e., object files, archive libraries and binary executables.

SCALE requires several TPLs, namely: QT and Lapack. These TPLs must be specified at the time of configuration. For ease of use, configuration scripts for every supported platform are available in the 'scripts' directory of the source code. These scripts describe the necessary variables to define. A user's modifications to these scripts should be limited to the path to the root directories for the TPLs.

Recommended Configuration Procedure

After unpacking out the source code, navigate to the root scale directory (<SCALE_SRC_ROOT>). You should see CMakeLists.txt, PackagesList.cmake and CTestConfig.cmake. This is the root of the source tree, what you will point CMake at. This example will demonstrate creating build trees for multiple configurations for your working copy.

Linux and Mac Configuration

- **Make Build Directory**

```
mkdir build
```

```
mkdir -p build/intel **this could be any directory
```

or

```
mkdir -p build/gcc
```

- **CMake Initialization**

Copy the cmake script from the 'scripts' directory to your build directory. Update the cmake script with your TPL specifications.

- Serial SCALE (without MPI support)

```
cp <SCALE_SRC_ROOT>/script/configure_scale_gnu.sh
build/gcc
```

```
chmod u+x build/gcc/configure_scale_gnu.sh
```

- Parallel SCALE (with MPI support)

```
cp script/configure_scale_mpi.sh build/gcc
```

```
chmod u+x build/gcc/configure_scale_mpi.sh
```

- **Create Your Configuration**

Create your configuration by pointing this script at the source tree root

```
cd build/gcc
```

Edit the CMakeLists.txt file in
<SCALE_SRC_ROOT>/Trilinos/packages/anasazi/src/ so as
to comment out the following line:

```
ASSERT_DEFINED(Anasazi_ENABLE_Tpetra)
```

- Serial SCALE (without MPI support)

```
./configure_scale_gnu.sh ../..
```

- Parallel SCALE (with MPI support)

```
./configure_scale_mpi.sh ../..
```

```
/** Configuration Output... **/
```

****Be aware:** if you add or remove source files from the source tree, CMake will NOT see these modifications unless a CMake file is modified (CMakeLists.txt, PackagesList.cmake, etc...). CMake will re-evaluate the entire source tree when any CMake file has been modified. If CMake does not pick up the addition/removal of sources files, the easiest way to update your build tree is to 'touch' any CMake file in the source tree.

Windows Configuration

Instructions are not documented here. Please contact scalehelp@ornl.gov for assistance.

Compilation

Every library and executable is a TARGET. Calling 'make' on Linux and Mac and 'nmake' on Windows from the root of your build tree

(<SCALE_SRC_ROOT>/build/gcc from the previous example) will build ALL targets. There are two options when building specific targets. For example MavricUtilities contains a number of executables: mtadder, mtaverager, and so on.

You may invoke 'make mtadder' to build mtadder alone. Alternatively you can, from build/gcc, 'cd packages/MavricUtilities' and 'make', which will compile ALL targets in mavricUtilities.

Compilation Flags

You may modify CMAKE_Fortran_FLAGS, CMAKE_C_FLAGS, CMAKE_CXX_FLAGS in on the cmake command invocation line.

Installation

CMake provides the 'install' target, which installs all binaries from the current directory down. For example, 'make install' from build/gcc will install any targets, which would install aim, mavric, mtadder, mtaverager, etc. You can also 'cd packages/Mavric' and 'make install' will only install targets declared in the Mavric directory.

***Things to know are that 'make install' re-evaluates the build for all dependencies. Take for example package A, which depends on packages B and D. Package D depends on package E. 'make install' for package A would result in packages E, D, and B being re-evaluated and rebuilt if necessary. Thus, if you know you want to build and install, you can save time building by simply doing a 'make install'. An alternative is 'make install/fast' which will skip the dependency checking and only execute the install operations.

References

- 1) W. J. Marshall, D. Wiarda, C. Celik, B. T. Rearden, and D. R. Wentz, “Validation of Criticality Safety Calculations with SCALE 6.2,” *Proceedings of ANS NCSD 2013 - Criticality Safety in the Modern Era: Raising the Bar*, Wilmington, NC, September 29–October 3, 2013, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2013).
- 2) T. Ueki and F. B. Brown, “Stationarity Modeling and Informatics-Based Diagnostics in Monte Carlo Criticality Calculations,” *Nucl. Sci. Eng.*, **149**, 38–50 (2005).
- 3) N. Larson, *Updated User’s Guide for Sammy: Multilevel R-Matrix Fits to Neutron Data Using Bayes’ Equations*, ORNL/TM-9179/R8, Oak Ridge National Laboratory, Oak Ridge, TN, October 2008.
- 4) S. Hart, “Implementation of the Doppler Broadening Rejection Correction in KENO,” *Trans. Am. Nucl. Soc.*, **108**, 423–426 (June 2013).
- 5) B. Becker, R. Dagan, G. Lohnert, “Proof and Implementation of the Stochastic Formula for Ideal Gas, Energy Dependent Scattering Kernel,” *Ann. Nucl. Energy*, **36**, 470–474 (2009).
- 6) T. M. Evans, A. S. Stafford, R. N. Slaybaugh, and K. T. Clarno, “Denovo: A New Three-Dimensional Parallel Discrete Ordinates Code in SCALE,” *Nucl. Technol.*, **171**, 171–200 (2010).
- 7) A. M. Ibrahim, D. E. Peplow, J. C. Wagner, S. W. Mosher, and T. M. Evans, “Acceleration of Monte Carlo Criticality Calculations Using Deterministic-Based Starting Sources,” *Trans. Am. Nucl. Soc.*, **105**, 539–541 (2011).
- 8) A. M. Ibrahim, D. E. Peplow, J. C. Wagner, S. W. Mosher, and T. M. Evans, “Acceleration of Monte Carlo Criticality Calculations Using Deterministic-Based Starting Sources,” PHYSOR 2012, Knoxville, TN, April 15–20, 2012.
- 9) A. M. Ibrahim, D. E. Peplow, K. B. Bekar, C. Celik, D. Ilas, J. M. Scaglione, and J. C. Wagner, “Hybrid Technique in SCALE for Fission Source Convergence Applied to Used Nuclear Fuel Analysis,” in preparation for the 2013 Topical Meeting on Nuclear Criticality Safety (NCSD 2013), Wilmington, NC, September 29–October 3, 2013.
- 10) E. Gabriel et al., “Open MPI: Goals, Concept, and Design of a Next Generation MPI Implementation,” in *Proceedings, 11th European PVM/MPI Users’ Group Meeting*, 97–104, Budapest, Hungary (2004).
- 11) M. L. Williams, K. S. Kim, “The Embedded Self-Shielding Method,” PHYSOR 2012, Knoxville, Tennessee, USA, April 15-20, 2012.
- 12) C. M. Perfetti and B. T. Rearden, “Development of a SCALE Tool for Continuous-Energy Eigenvalue Sensitivity Coefficient Calculations,” *Proceedings of Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013 (SNA + MC 2013) La Cité des Sciences et de l’Industrie*, Paris, France, October 27–31, 2013.
- 13) *International Handbook of Evaluated Criticality Safety Benchmark Experiments NEA/NSC/DOC(95)03*, Organization for Economic Co-operation and Development—Nuclear Energy Agency (OECD-NEA) (September 2012).
- 14) M. L. Williams, F. Havlůj, D. Wiarda, M. T. Pigni, I. C. Gauld, “SCALE Uncertainty Quantification Methodology for Criticality Safety Analysis of Used Nuclear Fuel,”

Proceedings of ANS NCSD 2013 - Criticality Safety in the Modern Era: Raising the Bar, Wilmington, NC, September 29–October 3, 2013, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2013).

- 15) B. T. Rearden, K. J. Dugan, and F. Havlůj, “Quantification of Uncertainties and Correlations in Criticality Experiments,” *Proceedings of ANS NCSD 2013 - Criticality Safety in the Modern Era: Raising the Bar*, Wilmington, NC, September 29–October 3, 2013, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2013).