SANDIA REPORT

SAND2016-4153
Unlimited Release
Printed May 2, 2016

Sierra Release Notes Version 4.40

Sierra Development Team

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation,
a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's
National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

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

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@adonis.osti.gov
Online ordering:	http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.fedworld.gov
Online ordering:	http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2016-4153
Unlimited Release
Printed May 2, 2016

Sierra Release Notes
Version 4.40

Sierra Development Team

Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185

Abstract

This document contains release information for the Sierra product. These changes are for the 4.40 release.

This page intentionally left blank.

Contents

List of Figures

List of Tables

This page intentionally left blank.

1 New Capabilities

1.1 New Capabilities for Aero

A more efficient communication strategy in parallel assembly was implemented.

The ability to specify density using the Reynolds number and a length scale was added.

1.2 New Capabilities for Low Mach

1.2.1 Particle Region

Several particle-wall resuspension models were introduced in this release.

1.2.2 Post-processing

A flux post-processor option at boundaries was created. This uses a residual-based reconstruction approach that is internally consistent with the applied governing equations rather than being post-processed from nodal values. This can be applied to any equation, on any boundary, by adding the following line to the boundary definition:

```
Postprocess [Total|Diffusive|Advective] Flux of [Equation Name]
```

The total flux post-processing is available on all boundary types, but ability to separate advection and diffusion depends on the boundary type, based on whether the boundary equations calculate and store these quantities separately or not. For example, inflow boundary conditions allow splitting these terms, while open and fixed boundaries do not. If you use the split form on one of these boundaries, it will use the nodal value to get the boundary value, which will result in some degree of error in the post-processed quantity.

Not all options are supported either. The Continuity equation does not have a diffusive component, so `Postprocess Diffusive Flux of Continuity` will result in an error.

The use of this capability with the Species equation is still a beta capability.

The result of this post-processing will be written to two nodal fields on the relevant boundary, named based on the flux type, equation, and sideset (e.g. `bc_Total_Enthalpy_flux_surface_2` and `bc_Total_Enthalpy_flux_surface_2_mag`). One output field is a vector, the other is a scalar which is the outward flux magnitude.

Note: This post-processing method may have errors at nodes shared by two boundaries if both of the boundaries have non-zero fluxes.

1.2.3 Boundary Conditions

The ability to use external fields was added for more options in boundary conditions. The use of the `external field for [X] = [Field Name]` now registers the field so it can be used with coupling (Fuego-Aria) and now supports non-primitive variables for `X` (e.g. `convection_temperature` on a convective boundary).

1.3 New Capabilities for Structural Dynamics

1.3.1 Navy NMount support

Navy users need to use their own non-shareable NMounts, but analysts have access only to binary installation: libraries and executables. Navy users can now create "SalinasUser_Sub_Mount.o". We provide capability to:

1. Insert new object file into libselem.a
2. Relink Salinas using the updated library
3. Users can use this new NMount as 'nmount type 8'.

1.3.2 Wet Modes

Modes of immersed Structures can be computed using an incompressible fluid approximation.[1] A portion of the fluid must be meshed and analyzed as part of the solution. This is more fully documented in the theory manual, with a detailed example in the "HowTo" documentation.

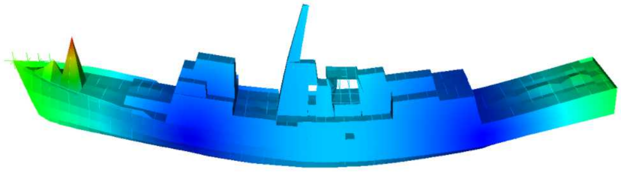

Figure 1: Wet Modes Calculation of a Ship

[1] In this approximation the eigen problem becomes,

$$\left(K_s - \lambda\left(M_s + LK_f^{-1}L^T\right)\right)\phi = 0$$

The approximation applies the effects of the fluid stiffness, K_s onto the structural mass surface.

1.3.3 Hydrostatic Balance of Meshed Rigid Bodies

Hydrostatic balance of rigid bodies, or “waterline” calculations may be followed by reposition of the body, elastic rebalance, or transient dynamics. This also includes dynamic relaxation as illustrated in Figure 2.

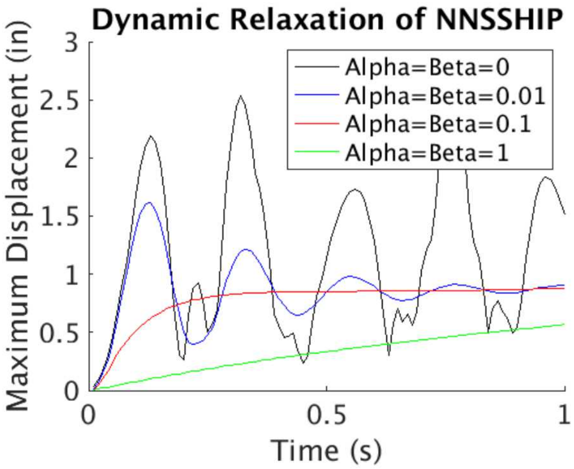

Figure 2: Dynamic Relaxation of Floating Ship Structure

1.3.4 Output of Layered Shell Stress

- Multi-layered shells may have stress output on each layer.
- Previous capability worked only in serial.
- Parallel capabilities are in place and tested for Nquad and Ntria.

1.3.5 Performance Evaluation: Evaluate SierraSD on platforms such as sierra133 using the haswell /Xeon architecture

This effort provides a baseline for future testing and evaluation as we move to new platforms. We found that Chama ran much faster for all problems than both the blade (baseline) and the Haswell platform. In general, the majority of the solution time (80% - 99%) is spent in the solver. On Chama especially, the majority of the solver time is spent in the solve phase (very little time is spent on initialization).

1.3.6 Memory Statics Output

By specifying the “mem_usage” keyword in the ECHO section, Sierra/SD now writes an additional file containing memory statics on a processor by processor basis. This additional output can be extremely helpful when memory usage limits the applicability of the solution.

1.3.7 Performance for Loading Vectors

We implemented a new sparse vector storage to improve memory and performance of multiple load vector applications. In some cases this can result in much improved memory usage.

1.3.8 Inverse Methods

The SD team has worked closely with the inverse methods team to fully deploy those methods that have been in the research phase for some time. Methods supported are shown in Table 1. All these methods are fully documented in User and How To documentation. Tests are in place for all supported capability, and the software has been refactored to improve software quality.

Method	Description
directfrf-loadid	Force identification in frequency domain
transient-loadid	Force identification in time domain
directfrf-materialid	Material identification in frequency domain
eigen-materialid	Material identification in Eigenvalue domain

Table 1: Supported Inverse Methods

1.4 New Capabilities for Solid Mechanics

Thermal and artificial material strains may now be defined in general directions. Previously these strains had to be either isotropic or aligned with the material coordinate system.

An automatic residual computation is now available in explicit quasistatic mode.

A user now has the ability to create an analytic object for contact. Currently, the only analytic object available is a plane.

A new element output variable called `cauchy_stress` has been added. The `cauchy_stress` is equivalent to the `unrotated_stress` rotated into the current global coordinate system. In contrast to the `stress` field, the `cauchy_stress` is defined for each integration point. For single integration point elements, `stress` and `cauchy_stress` are almost always equivalent.

Stress invariant output variables (`von_mises`, `max_principal_stress`, `triaxiality`, etc.) are now output at integration points for higher order and selective deviatoric elements. These values are now computed off of the `unrotated_stress` tensor. The only exception is `fluid_pressure` which is still computed using the `stress` tensor field for backward compatibility of certain features. Principal stress directions are now computed off of the cauchy_stress tensor field and are therefore also now output at integration point locations for higher order elements.

Added new derived variable outputs for max/min/intermediate principal strain directions for shells and bulk elements.

Initial overlap removal is now available for implicit analyses.

The EXTERNAL FORCE CONTRIBUTION OUTPUT NAME is now available in the fluid pressure and gravity boundary conditions. This command allows recording the external force produced by the boundary conditions in a separate nodal field.

If a path to a directory is specified as the location to write output (e.g. results files, history files, restart files, etc.) and the directories do not exist yet, they will now be created automatically behind the scenes.

The FLUID PRESSURE boundary condition can now accept an arbitrary FLUID SURFACE NORMAL direction, instead of being restricted to an axial direction.

New commands have been added to allow setting a contact specific shell thickness and lofting factor that overrides the default element mechanics lofting factor.

A new boundary condition called INERTIA RELIEF is now available for use. The INERTIA RELIEF command block computes a force or a moment that counteracts the external forces/moments in the problem. This capability is useful when a body is flying through space given an external force contribution, but the user would like the body to remain in a constant frame of reference.

The ability to compute the logarithm and exponential of global, element, and nodal fields has been added as a new user output reduction operation.

A new method ENFORCE ALL has been added for resolution of multiple MPC constraints. Additionally the meaning and behavior of the IGNORE option has been clarified.

A new boundary condition ARTIFICIAL STRAIN is available for applying artificial strain to groups of blocks. This method allows more flexibility and options than applying artificial strain in the material definition.

A new element type BOLT is available. The bolt model is a non-local beam element used for efficient representation of bolts and other fasteners in a finite element model. See the capabilities in development manual for details.

The inverse hyperbolic sine (asinh) is now usable from analytic functions.

A new derived output variable processor_id is available to view the parallel mesh decomposition.

The current cross sectional area of a beam can now be output via the current_area element variable.

1.5 New Capabilities for Thermal/Multiphysics

1.5.1 Porous Flow

Enable using either CVFEM or Galerkin FEM for `MASS_BALANCE` and `POROUS_ENTHALPY` equations. Galerkin FEM remains the default, CVFEM can be selected using the following solution options:

```
Begin Solution Options
  Begin Porous Flow Options
    Use CVFEM = true
  End
End
```

First order upwinding can be enabled when using CVFEM by using the material models below:

```
    Enthalpy Advection = porous_upwind
    Mass Balance Advective Flux of A = porous_upwind
```

1.5.2 Conservation Checking

The option to test for conservation of energy for basic heat conduction problems is now available. Use the line `Check Conservation of Energy` in your equation system to enable these checks, which will be printed to the log file. This option will not currently work when the advection or divergence terms are active.

1.5.3 Electrochemistry

The resistive load BC for the current equation now calculates the boundary area and calculates the flux in such a way as to be applicable to 2D and 3D problems.

1.5.4 Bulk Fluid Elements / Bulk Nodes

Multiple equations can now be solved on a single node. For example mass, energy, and species conservation simultaneously.

Boundary conditions coupling generic volume equations to bulk node equations can now be applied using the syntax:

```
BC BULKNODE_FLUX for VOLUME_EQUATION_NAME \
 [OF species_name] [IN material_phase] = \
 model_name BULK_NODE=bulk_node_name \
 [additional_model_parameters]
```

(The existing syntax for coupling convective and radiative flux boundary conditions to bulk nodes also remains valid for the Energy equation).

The new BC syntax supports coupling different volume and bulk node equations in cases where it makes sense. The rules for determining the bulk node DOF to couple the volumetric equation to are:

1. If the same DOF is present on both the volume and bulk node use that.
2. If the bulk node has the same DOF, but in the `NO_MATERIAL_PHASE` that is a valid candidate for coupling. For example the volume equation could be solving for the gas_phase pressure and that could couple to the pressure (with no material phase) equation on the bulk node.
3. Limited mappings between different degrees of freedom can be valid. For example an energy conservation equation solving for enthalpy can be coupled to one solving for temperature. The bulk node DOF can be in either the same material phase as the volume or in no material phase.

When not utilizing `IC READ_FILE` or `USE FILE VARIABLE`, allow use of a previous results file from analysis using bulk elements to be used as an grid file without having to modify the bulk element setup.

Allow the use of multiple bulknodes from an input mesh.

1.5.5 Pressurization Models

Pressurization models now automatically create a bulk node and solve mass and molar conservation equations on that bulk node. The mass and molar conservation equations on the bulk node can be coupled through boundary conditions to a porous media flow region. In the future it will be possible to apply boundary conditions directly to the bulk node, for example to model venting of a pressurized volume. At present the temperature of the pressurization model bulk node is still calculated using an averaging approach over the pressurized blocks. In the future there are plans to add an energy conservation equation to account for energy lost due to venting for example.

The mass and molar conservation equations can now be coupled to volumetric porous media conservation equations using appropriate boundary conditions that allow flow between the porous media domain and the pressurization model bulk node.

Initial support for modeling venting from a pressurization model has been implemented. Right now only a single model for the volumetric flow rate is provided, of the form $k*sqrt(P-P_{ambient})$. It can be enabled by adding the following line commands to the pressurization model block:

```
Venting Model = Vented
Venting Model Property Venting_Volumetric_Flow_Rate = K_FACTOR
Venting Model Property ambient_pressure = constant value = 13
Venting Model Property K_factor = constant value = 1.e-4
```

User functions, plugins, etc may be created for the `k_factor` or `venting_volumetric_flow_rate` if more sophisticated behavior is desired, and can be added to Aria as priority allows.

Multiple pressurization zones can now have their pressures coupled in addition to modeling venting to the exterior. This is accomplished by adding a `Bulk Node Coupling` block to the Aria Equation System (or Region if no equation systems are defined):

```
BEGIN Bulk Node Coupling coupling1
  Bulk Nodes = pZone1 pZone2
  Couple density model = k_factor_flow_with_choking critical_pressure_ratio=2.
  Couple species model = k_factor_flow_with_choking critical_pressure_ratio=2.
  Additional parameter k_factor = user_function name=k_coupling X=time
End
```

1.5.6 Laser Welding Boundary Conditions

Aria's laser welding boundary conditions have been updated to support 2 optional features. First, there is an optional parameter, `COMPUTE_VISIBILITY_FIELD`, for the `GAUSSIAN_SPOT_WELD` and `GAUSSIAN_LINE_WELD` BCs, for enabling a calculation of the visibility of the surface to the laser. Second, there is an optional parameter, `DEPTH_ENHANCED_LASER_ABSORPTION_COEFFICIENT`, for providing a table lookup for the absorption coefficient as a function of the relative surface depth and the aspect ratio of the surface. No longer is the `ABSORPTION_COEFFICIENT` given as a separate model. Instead these optional factors are multiplied by the optional `ALPHA` coefficient to give the overall absorption coefficient (default is 1.0).

1.5.7 Dirichlet Boundary Conditions

Enable use of single surface in defining Dirichlet BC from Field when the surface overlies multiple element blocks.

1.5.8 Anisotropic thermal conduction capability in Aria

Thermal conductivity in the solid phase can be prescribed as a summed contribution of the bulk conductivity and radiative conductivity. When prescribing conductivities of individual species, bulk conductivity can be specified as a tensor and supports scaling. All subsequent operations support tensors.

```
Begin Material Phase solid_phase
  ...
  tensor_bulk_conductivity_scaling of species1 = T_Exponent \
   kbulk_ref=0.5 T_ref=300. n=1.
  tensor_bulk_conductivity of species1 = constant  XX=1.0 YY=1.0 ZZ=1.0
  tensor_bulk_conductivity of species2 = constant  XX=2.0 YY=1.0 ZZ=1.0
  tensor_bulk_conductivity = volume_average
  tensor_thermal_conductivity = summed \
   contributions="tensor_bulk_conductivity radiative_conductivity"
End
```

The `Heat Conduction = Generalized` model now supports both scalar and tensor thermal conductivities, and should be preferred over `Basic` and `Fouriers Law` which do the same thing but only support scalar thermal conductivity.

1.5.9 Postprocessing

Support has been added for postprocessing the coordinates of the minimum/maximum temperature on a block using Encore. By adding the `output coordinates` line command to an encore min max postprocessor block a new global variable, `{postprocessor_name}_Coordinates`, will be created that stores the 3D coordinates of the minimum or maximum location. If both the minimum and maximum are being computed by the same postprocessor then the coordinates global variable will be of length 6 with the first 3 values corresponding to the coordinates of the minimum and the second 3 the coordinates of the maximum.

1.5.10 Phase Change Energy Source

Previously, analysts had to define a function of specific heat vs temperature for materials with multiple phase changes. The melting energy source expression has been updated so it can be used multiple times with different values for latent heat. This model provides better numerical performance than editing the specific heat function.

1.6 New Capabilities for Coupled Codes

This page intentionally left blank.

2 Changed Capabilities

2.1 Changes to the User Interface for Aero

2.1.1 Bug Fixes

- Fixed bug in SST turbulence model when using weak walls
- Fixed issue where initial condition for pressure would be wrongly computed if density was specified in the flow state
- Fixed initial condition from file for BDF2
- Fixed bug in run schedule: when `turb_conv_order` was set to 1 and comes before `spatial_order` set to 2, the turbulence convection order was not respected
- Fixed bug in heat flux output for when using a thermochemistry gas model

2.2 Changes to the User Interface for Low Mach

Transfer checking was added for particle simulations. At startup, the fields needed by the physics set in the particle region are compared with the fields actually being transferred from the fluid region. If fields are missing, an error will occur with a list of the missing fields. Expert users can add the flag `DISABLE PARTICLE TRANSFER CHECKS` in their particle region to convert these errors to warnings if desired.

Automatic activation of YPlus post-processing on all walls was added when a resuspension model is active in the particle region.

2.3 Changes to the User Interface for Structural Dynamics

2.3.1 Follower Loads: Ignore LOADS block for all multicase solutions

Previously, follower stiffness terms were computed during a 'receive_sierra_data' case by reading a LOADS section, but the LOADS section was ignored for multicase solutions, except in this case. The input was changed for consistency. Now the load section is referenced as part of the CASE, just as you would for any other multicase solution procedure. See the comparison in Figure 3. Pressure and Pressure_Z are the only supported follower loads. Attempts to call other load types as followers result in a fatal error.

```
SOLUTION
    case Preload
        receive_sierra_data
    case Modes
        eigen
END

LOADS
    sideset 1
        pressure = 2
        follower = yes
END
```

```
SOLUTION
    case Preload
        receive_sierra_data
        load = 1
    case Modes
        eigen
END

LOAD 1
    sideset 1
        pressure = 2
        follower = yes
END
```

Figure 3: Comparison of Old (left) and New inputs for follower loads

2.4 Changes to the User Interface for Solid Mechanics

Spot welds now function correctly with the rebalance capability.

The element field variables for tracking artificial and thermal strain have been consolidated now all element types use the variables: `nonmaterial_strain_3d`, `nonmaterial_strain_rate_3d`, `thermal_strain_rate_3d`, and `artificial_strain_rate_3d`. Additionally all these variables are now symmetric strain tensors, they use to be three component vectors.

The stable time step calculation for triangular shells has been updated to use a more efficient time step estimate. Additionally, a new time stable time step calculation option is available for quadrilateral shells.

The stable time step calculation for beam elements has been updated to correctly use offsets from the neutral axis.

The `CURRENT` effective moduli method is no longer available. It was determined that this method was untested and non-functional.

The default mass lumping algorithm for solid elements has changed. The new mass lumping scheme `ROWSUM` gives more accurate gravity loads and a more consistently stable explicit time step. There exists an option in the solid section block to force the code to use the old mass lumping algorithm for backwards compatibility.

By default rigid body history variables are now only output to results and history files if the analysis contains less than ten rigid bodies. If the analysis contains more than ten rigid bodies no rigid body history variables are output by default. This was done to prevent a large amount of history file bloat seen on analysis with many rigid bodies. The `EXTENSIVE RIGID BODY VARS OUTPUT` command can be used to change this default behavior.

The following material models were removed `COWPER_SYMONDS`, `HMR`, `HYDRIDE_PLASTICITY`.

Superelements may now be used in parallel. See the capabilities in development manual for details.

2.5 Changes to the User Interface for Thermal/Multiphysics

2.5.1 Specific Heat/Enthalpy Models

The default behavior for specific heat models is now to always create a compatible enthalpy model (if solving for temperature) or temperature model (if solving for enthalpy).

Using the generic polynomial or user function models for specific heat will automatically create an enthalpy expression if the variable listed is temperature, otherwise it will create the standard version.

2.5.2 Chemistry

The general chemistry `Time Function` option was changed to `Multiplier Function` and was extended to allow general functions of all variables, as well as the pre-defined functions for a tanh increase with either time or temperature as the dependent variable.

2.5.3 Rotated Boundary Conditions

Aria's handling of rotated boundary conditions has been updated to behave better in the case of multiple surfaces with rotated conditions applied to them meeting with a sharp angle at a node, as well as single surfaces with a rotated condition that contain faces which meet at a sharp angle. Previously the normal condition for every face touching a node was applied to a single rotated normal direction and both (in 3D) tangent directions were unaffected. The new behavior is to find an optimal rotated coordinate system that maximizes the alignment of all faces touching a node with one of the rotated axes and to apply the normal condition for each face to the rotated coordinate direction it is most aligned with.

2.5.4 Laser Welding Boundary Conditions

Add message to indicate deprecation of `LASER ENERGY FLUX` BC. Users should migrate usage of this BC to the more general `CIRCLE WELD ENERGY FLUX` BC.

2.5.5 KRINO/Level Sets

Krino has been refactored to use less memory for storing faceted surfaces. If the level set is initialized using a faceted surface, this facet file is still read in serial, but is immediately distributed in parallel to avoid inordinate memory usage on processor 0.

In order to get fully scalable memory performance for faceted surfaces and redistancing operations, the narrow band option must be used. This can be specified using 2 alternate line commands. To specify the absolute size of the narrow band size, the line command:

```
NARROW BAND WIDTH = 0.2
```

is used. This specifies that the signed distance should be calculated from -0.2 to 0.2, and clipped to -0.2 or 0.2 outside this range. This command is somewhat error-prone since it is easily copied from an old input deck to one where this length scale is no longer appropriate. The preferred usage is via a new line command:

```
NARROW BAND ELEMENT SIZE MULTIPLIER = 2.5
```

This specifies the narrow band range as a multiplier of the longest length scale of the elements of the problem at the time the mesh is read. For simplex elements, including triangles and tetrahedra, this is the longest edge of the problem. For quadrilateral and hexahedral elements, this includes the length of diagonals.

2.5.6 Contact

Shell lofting for contact was previously restricted to be from base of shell (0.0), midplane of the shell (0.5) and top of the shell (1.0). The range of shell lofting has been changed to lie anywhere in the range 0 to 1.

Bug fix for initialization of contact search in coupled simulations when read of a restart file is optional in the Aria region.

2.5.7 User Variables

In the past nodal element and scalar user variables could only be initially populated with constant values specified from the input file. Nodal and element user variables can now be populated with values defined on the input grid file.

2.6 Changes to the User Interface for Coupled Codes

2.6.1 Bug Fixes

Setting the number of parametric coordinates with shell element interpolation transfer was fixed.

This page intentionally left blank.

3 Issues Addressed

3.1 Issues Addressed for Structural Dynamics

Table 2 lists selected Structural Dynamics issues resolved in this release. For details or a full list, please contact sierra-help.

Table 2: Some of the Resolved Issues for Structural Dynamics

Summary	**User**
Scale factor applied multiple times to pressure load	Manktelow
default low frequency cutoff is incorrect for modal transient	Russ and Hopkins
its2sierra on SCN parsing and bad termination errors	Holzmann
different solutions for modal vs direct solves when mass attached with rbars	Hopkins and Owens
Pressure scaling with sideset variable seems off	Manktelow
slave_constraint_info isn't output for static solutions	Manktelow
How do I view AppliedPressure output variable?	Manktelow
F90 cray compiler issues	Davis
DirectFRF doesn't recognize non-integer frequency spacing	Schultz
rbar loading zero since 4.30 release	Miller
Parsing error with gravity applies loading in wrong direction	Kistler
Warning message for a negative surface jacobian	Manktelow
energy deposition reported is four times less than expected	Manktelow
How do I use the EDEP energy deposition variable input?	Manktelow
Negative start time should make run error out	Manktelow

This page intentionally left blank.

4 Issues with Backwards Compatibility

In this section we address changes in this release that may cause previous models to not run.

4.1 Incompatibility Issues for Low Mach

Rebalance was switched to STK - the imbalance threshhold parameter set in the particle region is now 0-based (rather than 1-based). Previously, to trigger a rebalance at 10% imbalance you would enter 1.1. Now you would enter 0.1.

The behavior of the legacy particle stick model was updated to be consistent with the new stick model (particles actually stick, and are no longer deleted).

4.2 Incompatibility Issues for Structural Dynamics

4.2.1 Memory Statistics Output

See section 1.3.6. A new file is now written.

4.2.2 Inverse Methods

Inputs designed for the previous *experimental* capability are not compatible with the production capability. See section 1.3.8. Backwards compatibility will not be maintained.

4.2.3 Follower Loads

Some consistency changes were made which may require an additional “load” specification in the “solution” section of the input. Details are in section 2.3.1.

4.3 Incompatibility Issues for Thermal/Multiphysics

4.3.1 Specific Heat/Enthalpy Models

The CpT enthalpy model has been removed. Using `Enthalpy = CpT` will result in an expression error.

The Temperature = linear_from_enthalpy model has been removed, and its use will result in an expression error

All the *_Evaluator specific heat models have been removed and rolled into the regular models. Using these models will result in an expression error.

4.3.2 Chemistry

Consolidated Chemeq_Rate_Multiplier with Reaction_Rate_Multiplier. If you were specifying the Chemeq rate multiplier in the material block, the syntax will have to change to Reaction Rate Multiplier = ... If you were specifying it in the ChemEq block itself (the more common case) no change is needed.

The general chemistry Time Function option was changed to Multiplier Function and was extended to allow general functions of all variables, as well as the pre-defined functions for a tanh increase with either time or temperature as the dependent variable.

4.3.3 Pressurization Models

The Initial mixture molecular weight line command is now required for all pressurization model blocks to determine the initial condition for the mass conservation equation that is now created.

4.3.4 KRINO/Level Sets

The syntax for specifying the velocity field associated with the level set interface has changed. This velocity field is only needed for advection by the krino region. So this should not be specified at all in aria. So the line:

```
Velocity Variable = solution->VELOCITY
```

in the Level Set interface block should be deleted.

For krino region usage, the level set used to advect the level set is now specified using the line:

```
EXTENSION VELOCITY = 0.0 0.0 1.0
```

This is specified in the Level Set interface block, not the region block.

4.3.5 ML Solver

There have been some instances noted where the ML solver will fail when used with contact and certain processor counts.

DISTRIBUTION:

1	MS 0899	Technical Library, 9536 (electronic copy)

Sandia National Laboratories