Release Notes for R12.1

November 16, 2021

Contents

§1 Release notes revision

This file constitutes revision 0 of the release notes for LS-DYNA version R12.1.0.

§2 License

The string ”REVISION 12” must appear in the LS-DYNA license file in order to run version R12.1.0. Please contact your LS-DYNA distributor or your Ansys sales representative if you have to have your license updated.

§3 Documentation

Documentation of R12.1.0 is provided in the R12.0 User’s Manuals which may be downloaded from `www.lstc.com/download/manuals`. For features mentioned in these release notes but which appear to be missing from the R12.0 User’s Manuals, please refer to the DRAFT User’s Manuals at `www.lstc.com/download/manuals`. Please note that not all features in the DRAFT User’s Manuals are available in version R12.1.0.

§4 Notes

The remainder of this file provides a brief description of new features, enhancements, and bug fixes in version R12.1.0. Some of the bug fixes may have also been included in other recent releases. Many of the new features and enhancements are also in R13.0.0.

The items are arranged by category. Understand that in many cases, a particular item may pertain to more than one category. In the interest of brevity, each item is listed only one, under a single category.

§5 New

§5.1 Constraint Cards

1. First experimental version to support solid element parts connected by ***CONSTRAINED_SPR2** or by ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3).

§5.2 EM (Electromagnetic Solver)

1. Added support for ***EM_RANDLES_EXOTHERMIC_REACTION** to batmac (***EM_RANDLES_BATMAC**).

§5.3 Miscellaneous

1. Add two new user subroutines,"`uinstability`" and "`udamage`", to connect with the properties/labels "Instability" and "Damage" of ***DEFINE_MATERIAL_HISTORIES**.

2. Add new user interface to define number of history variables on per part basis. We added new subroutine "`umat_setnhvpp`" to `dyn21umat.F`. It is called during initialization.

§6 Improvements and Enhancements

§6.1 Acoustics, NVH, and Frequency Domain

1. Implemented the output of `nodout_ssd` (***DATABASE_FREQUENCY_ASCII_NODOUT_SSD**) for direct SSD (***FEREQUENCY_DOMAIN_SSD_DIRECT**).

§6.2 Adaptivity

1. Add limited support for LSDA based keyword input files in adaptive models. These should be okay provided they do not contain ***NODE** or ***ELEMENT** data.

2. Added support for the file `INTFOR` during in-core adaptivity.

3. Numerous improvements to in-core adaptivity and dynamic load balancing.

4. Added an error message for when the initial mesh of 3D adaptive parts (ADPTYP = 7 in ***CONTROL_ADAPTIVE**) is not tetrahedrons.
5. 3D adaptivity now supports multiple ***CONSTRAINED_BEAM_IN_SOLID** definitions.
6. 3D adaptivity now supports ***CONSTRAINED_BEAM_IN_SOLID_PENALTY**.
7. Support of IREFLG for in-core adaptivity, which allows mesh to be homogeneously refined with in-core adaptivity.
8. Load curve, LCADP, used by 3D r-adaptivity is now output to `d3hsp` if it is nonzero.

§6.3 ALE

1. For ***BOUNDARY_AMBIENT** AMBTYP = 5 (***LOAD_BLAST_ENHANCED**), we've made the following enhancements:
   - Support multiple detonation sources
   - Obtain element density from ***LOAD_BLAST_ENHANCE** for better accuracy
   - Obtain nodal velocity from the area-weighted average of the segment velocity squared instead of segment velocity. This better conserves kinetic energy.
2. Extended the feature invoked with CNTTYP = 1 of *INITIAL_VOLUME_FRACTION_GEOMETRY to fill ALE elements with the material of a solid part in 3D (or shell part with 2D). The boundaries of the part specify the geometry of the container to be filled. The material in the part can optionally fill this container. This feature is useful for setting up an ALE model with a complex dummy structure that outlines a volume of ALE elements to fill.

§6.4 CESE Compressible Fluid Solver

1. Removed the CFL < 1.0 limit. Note that there is no such limit in the dual CESE solvers.

§6.5 *CHEMISTRY

1. Added code to properly deal with missing transport data in chemistry models when using ***CHEMISTRY** with ***CESE**.

§6.6 Constraint Cards

1. Introduced warning for bad definitions of ***CONSTRAINED_INTERPOLATION**, like for instance if independent nodes are co-aligned and thus unable to represent a rotation with respect to the corresponding axis.

2. PID1 and PID2 on ***CONSTRAINED_INTERPOLATION_SPOTWELD** can now refer to part sets, not only single parts. Set to negative part ID number to do so, that is, PID1 < 0 and/or PID2 < 0. This could be used for in-plane composed sheets, such as Taylor welded blanks.

3. Improve two rarely occurring issues with ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3):

   - More accurate search of associated shell element nodes, helpful in cases where shell elements are bent around corners.
   - Better solution for rotation vector if both normal vectors are colinear.

4. ***CONSTRAINED_TIED_NODES_FAILURE** now supports damage values from ***MAT_107** and ***MAT_110** to be used as failure variable for EPPF.

5. Specify whether ***CONSTRAINED_BEAM/SHELL_IN_SOLID** are the PENALTY version in the `d3hsp` file.

6. Implemented the start and end control (fields START and END) for ***CONSTRAINED_BEAM_IN_SOLID** and ***CONSTRAINED_SHELL_IN_SOLID** in the same way as for ***CONTACT_OPTION**.

§6.7 Contact

1. MPP improvements for beam handling in ***CONTACT_AUTOMATIC_GENERAL**

2. Fixed speed issue with MPP eroding contact on large core counts.

3. Update the option TETFAC on optional contact Card F to support 10-noded tets.

4. Improved the behavior of segment-based contact (SOFT = 2) in implicit solutions. Problems should converge better.

5. Backport of adding support to segment-based (SOFT = 2) contact friction for the viscous frction parameters VC on ***DEFINE_FRICTION**.

6. Enabled PSTIFF on ***CONTACT** to work with contact birth time. This option is for segment-based contact.

7. Modified how friction parameters are used in segment-based contact when ***PART_CONTACT** is used. If a segment pair in contact has nonzero friction parameters from only one of the segments, the nonzero friction parameters will be used for the pair. If friction parameters are defined for both parts, then the average is used. Previously, the average was used without checking values, so zero values could be used in the average causing the Coulomb coefficient to by 1/2 of the expected value.

8. Enabled FRCFRQ on Optional Card A of ***CONTACT_OPTION** to work with MPP contact when SOFT = 0 or 1.

9. Enabled ***ELEMENT_SHELL_SOURCE_SINK** to work with segment-based contact.

10. For segment-based contact, added option to omit the friction coefficient data from either the slave or master side. The Coulomb friction constant will be calculated using the coefficients for the other side.

11. Added support for TIEDIED = 1 on Optional Card D for ***CONTACT_TIEBREAK_NODES_TO_SURFACE**. TIEDIED = 1 enables incremental displacement update of the contact normal. Support was added for this contact to fix spurious stresses even with no loading.

12. Add option TETFAC to optional Card F on ***CONTACT_OPTION** to scale the computed volume of tet elements for the mass calculation in SOFT=2 contact. Better agreement is achieved with TETFAC between 3 and 5 when contact forces are compared between hex and tet meshes. This input field is supported for 4-node and 10-node tets.

§6.8 Control Cards

1. Staged construction (see ***DEFINE_STAGED_CONSTRUCTION_-PART** and ***CONTROL_STAGED_CONSTRUCTION**) did not work for thick shells (***ELEMENT_TSHELL**) with ELFORM = 3 or 5. Previously, the staged construction keywords had no effect on these elements, meaning they were always active.

2. Account for added mass in gravity load (see EMSCL in ***CONTROL_TIMESTEP**.

3. Added new keyword ***CONTROL_CONSTRAINED** for setting constraint related properties. Currently it only sets the behavior of the initialization check for SPR2 / SPR3.

4. Add new detailed materials energy option MATEN on ***CONTROL_ENERGY**. Currently this feature is supported for material models 3, 4, 15, 19, 24, 63, 81, 82, 98, 104, 105, 106, 107, 123, 124, 188, 224, 225, 240, and 251 for shell and solid elements. Setting MATEN = 2 causes the detailed material energies to be computed. For this calculation, internal energy is split into the elastic, plastic, and damage portions. They are computed and reported as "`mat_energy_elastic`", "`mat_energy_plastic`", and "`mat_energy_damage`" in `glstat` and `matsum`

5. Add option UMCHK on ***CONTROL_MAT** to potentially issue an error if only unmodified user material subroutines are called.

6. Add new option LISPSID on ***CONTROL_SHELL** to define a part set, for which the stresses from ***INITIAL_STRESS_SHELL** are with respect to the local coordinate system.

7. Added new option ENDMAS < 0 on ***CONTROL_TERMINATION**: A curve with ID = |ENDMAS| provides ENDMAS as a function of total mass.

8. Support ICOH = 1 of ***CONTROL_SOLID** for ***MAT_169**. This is automatic deletion if neighboring elements fail. It is already available for cohesive elements but now also available for this special material.

§6.9 Dual CESE Compressible Fluid Solver

1. Extended ***MESH** capabilities to permit use of ***MESH_BL**, etc.

§6.10 Elements

1. If the shell reference surface is offset by NLOC (***SECTION_SHELL**) or OFFSET (***ELEMENT_SHELL**), the time step size of those shell elements is reduced to fix instabilities. The reduction of the time step size is based on numerical tests which show a dependence on the offset distance and the ratio of shell thickness to edge length (T/L). Instabilities were most likely observed for aspect ratios of T/L>0.5.

2. Enabled thick shell formulation 3 to work with material models that use an equation of state.

3. Enabled the ORIENTATION option of ***ELEMENT_BEAM** to work with beam form 9, spot weld beams.

4. For ***ELEMENT_DIRECT_MATRIX_INPUT**:

   - Enhanced the reading of superelement input files to support connections using nodes, scalar nodes, and internal degrees-of-freedom. Also accept degree-of-freedom labels of ″0″ for scalar nodes.
   - Modified MPP decomposition logic for collecting nodes associated with superelements when reading binary formatted files for superelements.

5. Added the possibility to define 3-Point Simpson integration rule through the thickness per layer of a composite shell defined via ***PART_COMPOSITE**. This can be defined via an irregular optional Card 2 (OPTCARD).

6. Modified characteristic length for solid element formulation 4 to be more conservative in terms of computing added mass for enhanced stability and robustness.

7. Add new keyword option MISC for ***SECTION_SHELL** with first new parameter THKSCL. This allows the scaling of shell element thickness for all elements in the corresponding section at once.

8. Added new keyword option MISC for ***SECTION_SOLID**. The first new parameter is cohesive thickness, COHTHK. This allows section / partwise definition of cohesive thickness. This value supersedes the one from ***MAT_240** or ***MAT_ADD_COHESIVE**.

9. Added element quality check for solid cohesive elements (***SECTION_SOLID**, ELFORM = 19 / 20) to detect inverted elements in the first cycle.

10. IEBT = 3, coupled FEM / EFG method, on ***SECTION_SOLID_EFG** is now the default.

§6.11 Forming Analysis

1. Improved mapping algorithm for ***INCLUDE_STAMPED_PART**.

2. The new implementation of ***DEFINE_CURVE_STRESS** allows you to change the number of output stress-strain pairs based on the value of LCINT in ***CONTROL_SOLUTION**.

§6.12 ICFD (Incompressible Fluid Solver)

1. Added one and two parameter Pierson-Moskowitz spectrum for irregular wave generation invoked with WTYPE = 6 and 7, respectively, on ***ICFD_BOUNDARY_FSWAVE**.

§6.13 Implicit (Mechanical) Solver

1. Enabled additional keyword controls (Optional Card 3) from ***CONTROL_IMPLICIT_EIGENVALUE** for MPP execution. These controls were initially only available for SMP.

2. Enhanced ***PART_MODES** to work in single precision.

3. Some contact types were being turned off for the inertia relief computation invoked with ***CONTRO_IMPLICIT_INERTIA_RELIEF**. These contacts are now enabled.

4. For ***CONTROL_IMPLICIT_EIGENVALUE**, rewrote how data is output to the `d3plot`, `d3eigv`, and similar files, to vastly improve performance of the output. This saves 15

5. Allow shell element 18 to be used in modal dynamics (***CONTROL_IMPLICIT_MODAL_DYNAMICS**) when reading the modes from `d3eigv`.

6. Enhance the implicit solver's processing of nodal mass matrices to include the local coordinate transformation.

7. For intermittent eigenvalue analysis the direct solvers used by the eigenvalue analysis were forced for the entire execution. We now save and restore the solver option selected by the user to use during the non-eigenvalue execution of the run. This enables the use of iterative solvers.

§6.14 Initial Cards

1. Enabled ***INITIAL_STRESS_SOLID** to work when the data has more points than the element form has. For example, the data has stress data for 8 stress points, but the element has only 1 or 2 integration points.

2. Enabled ***INITIAL_VELOCITY_GENERATION** to work with solid element forms 16, 17, 45, and 61. Only the first 8 nodes were being initialized.

3. Added support for initializing solid elements with ***INITIAL_HISTORY_NODE(_SET)**. So far ELFORMS = -2, -1, 1, 2, 10, and 13 are supported.

§6.15 Isogeometric Analysis (IGA)

1. Make consistency flag, CONST in ***CONTROL_PARALLEL**, available for *IGA keywords.

2. Enable restart possibility for IGA elements.

§6.16 Load Cards

1. Fixed inconsistent behavior for ***LOAD_THERMAL_VARIABLE_NODE** in case of dynamic relaxation. Now we accept that the keyword is defined twice, where in one definition the load curve is active during DR and in the other it is active during the transient phase.

§6.17 Materials and Equations-of-State

1. ***MAT_ADD_COHESIVE** now supports hyperelastic foam materials ***MAT_057** and ***MAT_083**.

2. Support VOLFRAC in ***MAT_ADD_EROSION** so that it works correctly with failure criteria in ***MAT_ADD_DAMAGE_DIEM/GISSMO** and ***MAT_ADD_EROSION** itself. Before it only worked with failure in materials, such as FAIL in ***MAT_024**.

3. Enable GISSMO to be used with ***DEFINE_HAZ_PROPERTIES**. It is necessary to use ***MAT_ADD_DAMAGE_GISSMO_STOCHASTIC** to make it work.

4. Allow large function IDs $(> 2^24)$ for HISi < 0 and Dij of ***MAT_ADD_GENERALIZED_DAMAGE**.

5. Improved robustness if DMGEXP < 1 is used for ***MAT_ADD_GENERALIZED_DAMAGE**.

6. Extensions for ***MAT_ADD_GENERALIZED_DAMAGE** with IFLG2 = 1. It now supports the following additional material models for solid elements: ***MAT_133**, ***MAT_199**, and ***MAT_233**.

7. Properly support ***MAT_024** option VP = 3 in implicit analysis for solid and shell elements.

8. LCSS of ***MAT_024** with VP = 3 now supports ***DEFINE_TABLE_{*X*}D / COMPACT** up to a level of 9, meaning yield stress can be a function of plastic strain, strain rate, and up to seven history variables. If ***MAT_ADD_DAMAGE_DIEM** is used together with ***MAT_024**, then P1 of ***MAT_ADD_DAMAGE_DIEM** also supports a table up to level 9.

9. Output "maximum effective strain" as history variable number 29 for ***MAT_034** (FORM = 14, -14, and 24).

10. Added curves/tables with respect to plastic strain for temperature and transition stress parameters to ***MAT_30**.

11. Add ***MAT_036** as potential weld partner material for PROPRUL = 2/3 of ***DEFINE_CONNECTION_PROPERTIES**.

12. For ***MAT_068**:

    - Added velocity dependent hardening. Hardening curve IDs LCPD{R,S,T} and LCPM{R,S,T} can now refer to tables, meaning velocity dependent yield forces / moments can be defined.
    - Add velocity dependent failure. Negative input of {F,M,U,T}FAIL{R,S,T} can now refer to curves, meaning velocity dependent failure forces / moments / displacements / rotations can be defined.

13. Fully iterative plasticity scheme implemented for ***MAT_SIMPLIFIED_JOHNSON_COOK** (***MAT_098**). It should improve robustness.

14. Add keyword option UNIAXIAL for ***MAT_100** and ***MAT_100_DA**. In addition to setting the transverse shear stresses to zero (E < 0), the transverse shear strain increments get zeroed as well. This can reduce occasional instabilities.

15. For ***MAT_100_DA** / ***MAT_SPOTWELD_DAIMLERCHRYSLER**:

    - Improved the material tangent in the plastic regime to yield better convergence in implicit analysis.
    - Added rate filtering option with new parameter RFILTF on card 3, entry 6.

16. Implemented erosion flag for ***MAT_JOHNSON_COOK** / ***MAT_110** for shells.

17. Added nonlinear viscoelastic model to ***MAT_123** (VP = 1) for shell and solid elements. Viscoelastic approach adopted from ***MAT_187** with LCEMOD > 0.

18. Add new option SRFILT to ***MAT_124**. This enables strain rate filtering using exponential moving average. It works for LCSRC or LCSRT being nonzero and SRFLAG = 0 or 1.

19. Reimplementation of plasticity algorithm for ***MAT_136** to improve response for kinematic and mixed hardening.

20. Added option to specify the energy release rates for modes I and II (GIC and GIIC) in ***MAT_138** as element size depende4tn curves.

21. Backport of implicit tangent for ***MAT_SAMP_LIGHT** / ***MAT_187L**.

22. Add new parameter RATEOP to ***MAT_SAMP_LIGHT** (***MAT_187L**) to choose type of effective strain rate calculation.

23. Improvements to ***MAT_SAMP_LIGHT** (***MAT_187L**) to better deal with highly nonlinear inputs, meaning sharp curves and logarithmic strain rate interpolation.

24. For ***MAT_BOLT_BEAM** / ***MAT_208**, added feature (see input parameter HOLSHR) to enable modeling of a shear deformation mode in which the bolt shank tears through the plates. After the clearance gap has been closed, further shear deformation would enlarge the hole in the plates rather than deform the bolt itself. The force-deformation relation of this mechanism is still governed by LCSHR,

but the deformation (i.e. enlargement of the hole) is tracked separately in each of four principal directions. Thus, enlargement of the hole in the positive Y-direction has no effect on the position of the edge of the hole in the negative Y-direction.

25. Align element failure behavior in ***MAT_224_GYS** with the one in ***MAT_224**, e.g., NUMINT=-200 should prevent element erosion at all.

26. Added keyword option FUNCTIONS to ***MAT_240** to include connection partner properties. With FUNCTIONS, the following parameters can/must be defined as ***DEFINE_FUNCTION** IDs: EMOD, GMOD, G1C_0, T0, FG1, G2C_0, S0, and FG2 (and GMOD3, G3C_-0, R0, and FG3 if keyword option 3MODES is also used). The arguments of these functions include thicknesses and maximum yield stresses of both weld partners (from tied contact) as well as the strain rate and element area. A similar option is already available for ***MAT_169**.

27. Add new option for yield/damage initiation criterion of ***MAT_240**. Parameter INICRT can now be given as negative value, invoking a flexible exponent in the yield and damage initiation criteria.

28. Added optional rate filtering method RFILTP to ***MAT_240**: exponential moving average of the effective plastic strain rate (RFILTP > 0) or of the effective total strain rate (RFILTP < 0).

29. Add two new history variables for *MAT_240, namely number 16 is FG1 and number 17 is FG2.

30. Support ***MAT_251** in implicit analysis for solid and shell elements.

31. Add ***MAT_251** as a potential weld partner material for PROPRUL = 2 / 3 of ***DEFINE_CONNECTION_PROPERTIES**. Therefore use values from constant inputs PHASE1 and PHASE2 to find the associated curve in a ***DEFINE_TABLE_3D** or ***DEFINE_TABLE_4D**.

32. For ***MAT_254**, moved extended Koistinen-Marburger model to a new transformation law 9. Transformation 1 is the standard Koistinen Marburger again to ensure backward compatibility.

33. For ***MAT_261** (***MAT_LAMINATED_FRACTURE_DAIMLER_-PINHO**):

- Check for fiber kinking, even if fmat = 1.0. This allows for proper behavior if fiber compression and matrix tension are present at the same time.
- Added minimal values for fracture toughnesses if not defined to avoid division by zero.

34. For ***MAT_262** (***MAT_LAMINATED_FRACTURE_DAIMLER_CAMANHO**), add a warning message in case SL > YC, as this may lead to unexpected behavior especially in fiber compression. IF SL is big enough, the approximation for ″etal″ (longitudinal friction coefficient) will result in values >1. That will then in turn lead to phifc always being 0 (eq.(21) in CamI-paper).
35. Add new options(s) to ***MAT_280**: FMOD = 10, 11, and 12 use a slightly modified condition for compressive failure.
36. Add new option to define rate dependent strength in ***MAT_GLASS** (***MAT_280**). It is invoked by FT < 0, with curve ID |FT| specifying tensile strength as a function of strain rate.
37. New case for ***MAT_280** to allow FT < 0 to be used with FTSCL > 0. In that case, |FT| defines a curve for tensile strength as a function of strain rate and FTSCL scales the strength values from that curve as long as the material is intact. If cracked, neighbors get non-scaled values from that curve.
38. Minor change for ***MAT_280** with ENGCRT and RADCRT. If used together with ***MAT_ADD_EROSION** (where these criteria are also available), ***MAT_280** only ″wins″ if those parameters are nonzero.

§6.18 MPP

1. Added keyword ***CONTROL_MPP_REBALANCE** to control features associated with dynamic load balancing.

§6.19 Output

1. Move `dyna.lsda0000` file from the ″local″ to the CWD. This change allows you to not worry about copying it over from the local directory when you are using ***CONTROL_STRUCTURED**.
2. Field NPLTC on ***DATABASE_BINARY_OPTION** works for keyword options INTFOR, D3DUMP, and DEMFOR.

3. For ***DATABASE_RECOVER_NODE**, added support for solid element formulations - 1 and -2 for the elemental extrapolation method (METHOD = 1).

4. Enabled output of moments for 2D plane strain elements to the section force file, `secforc`.

5. Added forces from ***LOAD_ERODING_PART_SET** to `nodfor` output.

6. Added alternative way to specify circular cross section for ***DATABASE_CROSS_SECTION_PLANE**. When RADIUS < 0.0, the radius is the absolute value of RADIUS, and XCT and XCH are node IDs. The center of the circular cut plane is given by the location of XCT. The normal vector of the plane is the vector pointing from XCT to XCH.

7. Support output of plastic strain tensor for solid element 10 (see STRAIN on ***DATABASE_EXTENT_BINARY**.

8. Added new field DELERES to ***DATABASE_EXTENT_BINARY** to write last available results to `d3plot` and `d3part` for deleted elements.

9. Add two new quantities for SPR3 (***CONSTRAINED_INTERPOLATION_SPOTWELD**) to `binout` / `swforc` database: "resultant_axial" (manual: F_n) and "resultant_shear" (F_s).

10. Added "state variable" of ***CONSTRAINED_INTERPOLATION_-SPOTWELD** (SPR3) to `binout` / `swforc` output.

11. Print header text of ***DEFINE_CURVE_TITLE** and ***DEFINE_COORDINATE_..._TITLE** to `d3hsp`.

12. Add echo of command line options to screen, `d3hsp`, and `messag` files.

§6.20 XFEM (eXtended Finite Element Method)

1. Added jobid to "cracks" and cracktip files generated by XFEM simulation.

§6.21 Miscellaneous

1. Enable support for the new ***VENDOR_LICENSE** scheme, both with the FlexLM server and with the lstc_server licensing system. Documentation for this will be released soon.

2. Significantly sped up some load curve evaluations, particularly if there are many points in the curve definition, by changing the search from linear to binary.

3. Added new sense switch swe. This new sense switch (swe.) will force an explicit dynamic relaxation solution to immediately stop as though converged, write the corresponding `d3dump01` data, and then commence with the regular solution. Sometimes it's clear that the DR solution is very close to convergence but is making no further progress in terms of achieving the DRTOL criterion set in the input deck. In such a case, you will usually want to call the DR solution good as is.

4. Added new sense switch endtime=t. This sense switch changes the termination time.

5. Added flag, IMONFLAG, to field 5 of ***DEFINE_HAZ_TAILOR_-WELDED_BLANK** to allow load curves, ISW and IFW, from Card 2 of ***DEFINE_HAZ_PROPERTIES** to either increase monotonically (IMONFLAG = 0) or arbitrarily (IMONFLAG = 1).

6. Add new keyword ***DEFINE_TABLE_COMPACT**. It allows a more compact notation of 2D-, 3D-, 4D-, ... tables. Everything takes place in the keyword reader, where a ***DEFINE_TABLE_{*X*}D** is created internally.

7. Add new keyword option NAMES to ***DEFINE_MATERIAL_HISTORIES**. This will allow you to specify names for history variables.

8. Activate error / warning for failed / unconstrained spot weld assembly solids (***DEFINE_HEX_SPOTWELD_ASSEMBLY**). Up to now, this error or warning (depending on SPOTSTP from ***CONTROL_-CONTACT**) was only issued for single spot weld solids.

9. Extended the ***USER_LOADING** interface to support thick shell elements.

§7 Bug Fixes

§7.1 Acoustics, NVH, and Frequency Domain

1. Fixed a bug in converting pressure load to nodal force, for frf, ssd and random vibration (***FREQUENCY_DOOMAIN_FRF**, ***FREQUENCY_DOMAIN_SSD**, and ***FREQUENCY_DOMAIN_RANDOM_VIBRATION**). The results before this fix were incorrect when pressure loading a curved surface.

2. Fixed problems in accessing `d3eigv` with nonzero neips/neiph values. The eigenvectors were not extracted correctly before this bug fix.

3. *FREQUENCY_DOMAIN_RESPONSE_SPECTRUM Fixed a bug in combining ground motion in response spectrum analysis (***FREQUENCY_DOMAIN_RESPONSE_SPECTRUM**) when the load is defined by time history (LCTYP = 10, 11, or 12).

4. *FREQUENCY_DOMAIN_FRF Fixed bug for frf (***FREQUENCY_DOMAIN_FRF**) with response in all x, y and z directions. Before this fix, the output file names are incorrect.

5. Fixed a bug when running SSD (***FREQUENCY_DOMAIN_SSD**) in MPP which only shows up on IBM and Windows.

§7.2 Adaptivity

1. Fixed bug causing error message STR+389 (solid elements referencing a non-existing part) that occurred when using ***RIGIDWALL_GEOMETRIC_FLAT_MOTION_ID_DISPLAY** with adaptivity.

2. For ***USER_INTERFACE_FRICTION**, support h-adaptivity (MPP), meaning if ***CONTROL_ADAPTIVE** with ADPTYP = 1 or 2 is used, then history variables uhnew of subroutine `usrfrc` are transferred correctly from the old to new mesh in each adaptive refinement step.

3. Fixed bug in 3D r-adaptive parts that caused them to possibly lose feature lines after remeshing.

4. Corrected memory allocation for shell h-adaptivity (ADPTYP = 4 on ***CONTROL_ADAPTIVE**). The issue could lead to a segmentation fault.

5. Fixed bug in 3D r-adaptivity so the stresses in non-adapted parts are correct.

§7.3 Airbag Cards

1. Fixed bug associated with the airbag reference geometry which happened when a node not associated with any shell elements is specified in ***AIRBAG_SHELL_REFERENCE_GEOMETRY**. This bug could cause difficulties in the element normal calculations.

§7.4 ALE

1. Fixed bug for S-ALE mesh trimming (***ALE_STRUCTURED_MESH_TRIM**) for the case of trimming both sides of a structure by specifying a part set. Previously, only one side of the structure was being trimmed, even if both sides were requested.

2. Fixed error in SMP Windows version for S-ALE mesh trimming (***ALE_STRUCTURED_MESH_TRIM**) that may lead to NaN.

3. Fixed bug for S-ALE with ***MAT_HIGH_EXPLOSIVE_BURN** / ***MAT_008**. The bug caused an incorrect detonation time.

4. Bug fix for S-ALE with moving meshes in the MPP versions. The bug caused incorrect results.

5. Bug fix for ***ALE_STRUCTURED_FSI** with Lagrange structure containing triangles. It might cause erroneous results, mostly leakage. It sometimes caused crashes.

6. Fixed memory bugs for ***ALE_STRUCTURED_FSI**. The bugs were in certain cases causing the time step to drop or crashes.

7. Fixed issues associated with ALE/S-ALE 2D models that use ***BOUNDARY_NON_REFLECTING_2D**. The implementation was not accounting for all the 2D segments.

8. S-ALE with ***ALE_ESSENTIAL_BOUNDARY** did not work properly in R12.0. It is fixed in this version.

9. Fixed bugs for S-ALE with mass damping. The bugs were causing incorrect results.

§7.5 Boundary Cards

1. Fixed a bug for ***BOUNDARY_PRESCRIBED_MOTION_SET_BOX**. The time was not being offset correctly when TOFFSET = 1.

2. Fixed an MPP bug for ***BOUNDARY_PRESCRIBED_MOTION_SET_BOX**. It occurred when the node set is part of a rigid body.

3. Fixed bug that caused wrong imposed velocity when using DOF = 4 in ***BOUNDARY_PRESCRIBED_MOTION**.

4. Fixed bug that made ***BOUNDARY_PRESCRIBED_MOTION** not work with ***DEFINE_BOX_LOCAL**.

5. Fixed bug that caused wrong external work output in `GLSTAT` when there are ***BOUNDARY_PRESCRIBED_MOTION**'s that supersede ***BOUNDARY_SPC**'s.

6. Corrected a precision conversion problem associated with using superelements in explicit single precision with velocity boundary conditions.

7. For ***BOUNDARY_MCOL**:

   - Corrected `RBDOUT` velocities for MCOL rigid bodies. Prior releases of R11 and R12 saved incorrect velocities to `RBDOUT`. The actual solution results are correct.

   - Corrected an MCOL subroutine for wave effect damping. All prior releases which used more than 0 and less than 60 wave damping matrices in `MCOLFILE` calculated incorrect wave effect damping. This could result in unstable solutions. Solutions which did not include wave effect damping or which used exactly 60 wave damping matrices are unaffected.

8. ***BOUNDARY_NON_REFLECTING** now uses the correct velocity in implicit so that explicit results are matched.

§7.6 CESE Compressible Fluid Solver

1. Fixed an MPP `d3plot` output bug related to surface plot data.

2. Fixed a CESE solver cyclic boundary condition issue.

3. In the moving mesh FSI solver:

- Fixed some structural shell indexing.
- Corrected the nodal connectivity for wedge/prism elements.

§7.7 Constraint Cards

1. Fixed incorrect initial velocity of ***CONSTRAINED_NODAL_RIGID_BODY** that occurs when using ***INITIAL_VELOCITY_GENERATION** with NX = -999. The rotational axis was incorrect.

2. Fixed bug in gear joints (***CONSTRAINED_JOINT_GEARS**) for implicit. We were not accounting for the helix angle.

3. Fixed bug in Lagrangian multiplier approach of ***CONSTRAINED_INTERPOLATION**. It has been broken for some time before the release of R12.0.0.

4. Fix for ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3) with huge number of associated nodes. A memory issue was likely to occur if more than 200 nodes were found in the domain of influence of upper or lower shell element part.

5. Fix for ***CONSTRAINED_SPR2** and ***CONSTRAINED_INTERPOLATION_SPOTWELD** (SPR3): If nodes in the domain of influence get completely free due to parent element failure, these nodes should not get any forces or moments distributed anymore.

6. Fixed several bugs in implicit thermal analysis with ***CONSTRAINED_BEAM_IN_SOLID**:

   - Initialize temperatures at beam nodes, so correct initial temperatures are set in linear and nonlinear thermal analyses.
   - Correctly add constraints for temperature DOFs at beam nodes when a temperature gradient is calculated through the thick shell thickness (THSHEL = 1 in ***CONTROL_SHELL**).

§7.8 Contact

1. Fixed the contact energy density when segment-based contact is used (SOFT = 2 on Optional Card A of ***CONTACT**). The values were too low.

2. Fixed SOFT = 2 (segment-based) eroding contact when elements erode before the contact birth time. Segments were not being eroded.

3. Changes for MPP SOFT = 0/1 contact with birth time to check for any contact (shell) segments that may have failed before the contact became active. Previously, these checks only happened once the contact was active and so neglected to delete contact segments from shells that failed before the contact birth time. This fix applies to ***CONTACT_ATUOMATIC_SINGLE_SURFACE** and ***CONTACT_AUTOMATIC_*_TO_SURFACE**. It covers both groupable and non-groupable.

4. Fix for ***CONTACT_RIGID_SURFACE** initialization which has in some cases been using the wrong coordinates in the single precision version of the code since 08/10/2016.

5. Fix for MPP groupable drawbead (***CONTACT_DRAWBEAD**) behavior when the contact is off the edge of a thin triangle. It was producing NaNs during the friction calculation.

6. Fix for MPP groupable drawbead contact (***CONTACT_DRAWBEAD**). It had an overly restrictive bucketsort, so it could miss some of the ramp up force a drawbead should create.

7. Fix for cohesive element failure issue with MPP groupable tied contact. Groupable tied contact does not initialize until the first cycle, so checking the "is this node tied" flags before then was causing premature element failure.

8. Fix initial tying in MPP groupable tied contact which was not honoring the part specific thickness from ***PART_CONTACT** which could result in some nodes not being tied that should be.

9. Fix for MPP tied contact birth/death times when given by load curves.

10. Fix for MPP eroding contact to properly handle element deletion during small deck restarts.

11. Fix for race condition bug in MPP contact initialization, which could on rare occasions cause the code to hang.

12. Fix a potential MPP hang if the penalty backup contact (IPBACK on Optional Card E) is not assigned to the same processors used for the primary tied contact.

13. Fixed segment-based contact (SOFT = 2 on Optional Card A of ***CONTACT**) when used with PSTIFM > 0 and selective mass scaling. Segment masses were being scaled incorrectly causing excessive stiffness.

14. Fixed bug for single surface contacts when SSID = 0 and ***ELEMENT_SHELL_COMPOSITE** is in the input deck. Not all parts were being included.

15. Fixed a flaw in MPP segment-based eroding contact that could cause an increase in time for bucket sorts when elements in contact erode. It could also make bucket sorts slower when the REGION field on Optional Card E is used with any MPP segment-based contact.

16. Fixed using SBOXID and MBOXID with segment-based contact. A redundant calculation was causing contact to be lost.

17. Fixed ***TERMINATION_CONTACT** to make it work when used with THRES > 0 or DOF = 2 or 3. The threshold was being compared with the maximum nodal contact force instead of the resultant. For DOF = 2 and DOF = 3, the z and y force, respectively, were being checked instead of y and z. Now resultant force is used and all options are working.

18. Fixed force transducers when used with MPP contact with SOFT = 0 or 1. The fixed keywords are types are **AUTOMATIC_GENERAL**, **SURFACE_TO_SURFACE**, **NODES_TO_SURFACE**, and **ONE_-WAY_SURFACE_TO_SURFACE**. Forces were sometimes being assigned to the wrong interfaces or to nonexistent interfaces, in which case a memory error could have cause as segmentation fault.

19. Fixed force transducers when use with segment-based (SOFT = 2) groupable MPP contact.

20. Fixed MPP solutions with 2D force transducers (***CONTACT_2D_-FORCE_TRANDUDCER**) that have both a slave and master surface defined.

21. Fixed orthotropic friction in segment-based contact. It was not calculating the Coulomb friction constant as expected.

22. Previously ***CONTACT_AUTOMATIC_GENERAL** in SMP missed the contact between beams generated with ***ELEMENT_BEAM_-SOURCE**.

23. Fix ineffective ***CONTACT_TIED_SURFACE_TO_SURFACE** after dynamic relaxation when death time is set to -9999.

24. Fixed bug causing an input error saying the element has zero nodal thickness (STR+230) when using triangular elements with shell type 23 (8-node quadrilateral shell) for contact.

25. Fixed immediate core dump during runtime or memory error upon termination when using SOFT = 0/1 tied contacts with SOFT = 2 surface-to-surface and single surface contact and running with SMP.

26. Fixed incorrect duplication of higher order shell (23 and 24) parts for contact.

27. Fixed bug that was resulting in contact not being detected when the slave box (SBOXID) is defined with ***DEFINE_BOX_LOCAL** abd the master box (MBOXID) is defined with ***DEFINE_BOX** as specified on Card 1 of ***CONTACT_OPTION**.

28. Fixed segmentation fault when using ***CONTACT_SLIDING_-ONLY_PENALTY** and running with SMP and NCPU $\neq 1$.

29. Fixed bug in the SMP version of ***CONTACT_SURFACE_TO_SURFACE**. This contact is two-way, but one-way contact was being performed instead.

30. Fixed a bug during initialization for MPP when there were a large number of contact in the model. The bug caused a segmentation fault.

31. Extended the logic for collecting the resultant forces for the INTFOR file for ***CONTACT_TIED_SURFACE_TO_SURFACE** when using implicit in MPP. Before the resultant forces were being reported as zeroes for this contact type.

32. Added a warning that ***CONTACT_TIEBREAK_NODES_TO_SURFACE_ID** is not supported for implicit computations.

33. Fixed a bug in initial penetration check for triangular segments for ***CONTACT_AUTOMATIC_BEAMS_TO_SURFACE**. Prior to this fix, unreasonable penetration may have been computed that would have led to wrong contact forces.

34. Fixed memory clobber in ***CONTACT_ADD_WEAR** for MPP.

35. Fixed issue with failure and eroding Mortar contact in SMP. This issue occurred during the deletion of elements.

36. Fixed bug in Mortar contact when used with high order elements. The normalization normal vectors was missing which resulted in wrong results.

37. Fixed memory clobber when eroding contacts are used. It could result in segment violation on some platforms.

38. Avoid groupable for Mortar contact since it is not supported.

39. Fixed beam mass calculation for ***CONTACT_AUTOMATIC_-BEAMS_TO_SURFACE**. Beams with ***MAT_NULL** and RO < 1.e-11 get zero nodal mass, but that could lead to later problems in contact, now resolved.

40. Fixed input check for thermal contact with variable heat conductance (LCH in the cards for the THERMAL_FRICTION option of ***CONTACT**). Bug could result in neglecting heat transfer and throw a warning even if curve was correctly defined.

§7.9 Control Cards

1. For any kind of implicit eigenvalue computation, LS-DYNA previously ignored the value of TET10S8 on ***CONTROL_OUTPUT** and only wrote the corner nodes of the elements to `d3plot` for the connectivity. Now it accepts the setting of TET10S8, so the full connectivity can be written to the `d3eigv` database.

2. For stage construction, fixed bug related to unexpected forces being present in dormant elements. The bug affected solids (hourglass type 6 only), thick shells (all hourglass types), and shells (stiffness-method hourglass types). When a part is dormant (meaning the construction stage when the part becomes active has yet been reached), stress and forces are calculated but are scale down (see input parameter FACT on ***CONTROL_STAGED_CONSTRUCTION**). The scaling down had been erroneously omitted for the hourglass force pertaining to the element types and hourglass setting mentioned above.

3. The following fixes have been applied to pore fluid analysis (see ***CONTROL_PORE_FLUID**):

- For certain models the output to `d3plot` was wrong. The pore pressure variables could be written to the wrong position in the `d3plot` file, resulting in stresses or other extra history variables being overwritten with pore pressure, and the extra history variables that were expected to contain pore pressure data instead contained zero. The bug apparently occurred only with small simple models and the did not seem to affect large complex models.
- Bug affecting flow through tied contact (FTIED on ***CONTROL_PORE_FLUID**) together with suction limits (see SUCLIM on ***BOUNDARY_PORE_FLUID**). If a node on the tied contact reached the suction limit, the suction limit pressure then spread along the tied contact in a non-physical way.
- Bug affecting flow through tied contact (FTIED on ***CONTROL_PORE_FLUID**) together with time-dependent consolidation (ATYPE = 3). Pore pressures could become unstable at nodes shared across MPP processors where the node belongs to more than one tied contact.
- Bug affecting ATYPE = 4 (steady state consolidation), which could occasionally occur when thin elements were sandwiched between thick elements. The calculation became unstable in the first few time steps, with pore pressure oscillating between high positive and negative values. A work-around for the bug was to use the curve ACURVE on ***BOUNDARY_PORE_FLUID** to set the analysis type to time-dependent consolidation (ATYPE = 3) for at least 100 time steps before switching to ATYPE = 4.

§7.10 Dual CESE Compressible Fluid Solver

1. Corrected some issues with axisymmetric boundary conditions.
2. Corrected some bugs with wedge/prism elements.
3. In the moving mesh FSI solver:
   - Fixed some structural shell indexing.
   - Corrected a dual CESE internal boundary condition segment counter.

§7.11 Elements

1. Bug fix for PARM3 for discrete beams defined with ***ELEMENT_BEAM_THICKNESS**. The local coordinate ID specified by PARM3

was ignored by the discrete beam material models ***MAT_069**, ***MAT_070**, ***MAT_071**, ***MAT_074**, ***MAT_094**, ***MAT_197**, ***MAT_205** and ***MAT_208**. The global coordinate system was used instead.

2. The `d3plot` file was becoming corrupted when a 4-node element formulation was specified for 8-node shell elements. An error message is now issued when an improper element formulation is specified for 8-node shell elements.

3. Stresses where not correctly assembled in MPP for solid element formulation 13.

4. For ***ELEMENT_SEATBELT**:

   - Fixed a bug for 2D seat belts that can result in segment faults when there is no retractor in the model.
   - Fixed a 2D seat belt bug that occurred when a belt has two retractors.

5. Increased mass scaling for cohesive shell (ELFORM = 29 in ***SECTION_SHELL**).

6. Fixed shell element composite modeling when ESORT on ***CONTROL_SHELL** is set to sort triangular elements into triangle forms, and the sorting changes the number of in-plane integration points or the ordering of the data.

7. Fixed an error in the MCID keyword option of ***ELEMENT_SHELL**. A redundant conversion from degrees to radians was changing the material direction.

8. Enabled thick shell formulations 3, 5, and 7 to correctly initialize the pressure to zero when used with equation-of-state materials. There was some pressure due to round-off error.

9. Fix seg fault when writing out `d3dump` file after dynamic relaxation when using a large number of layers in ***ELEMENT_SHELL_COMPOSITE**. This affects MPP only.

10. Fixed bug causing spurious error message KEY+1567 ("ELEMENT ### is not found") when using ***ELEMENT_SHELL_COMPOSITE** and when ***MAT_ADD_THERMAL_EXPANSION** is present in the input.

11. Corrected the setting of nodal inertias for solid element formulation 3 for eigenvalue computations. It was zero, and now it is the standard computed value.

12. Corrected the computation of the critical time step for explicit using of superelements (***ELEMENT_DIRECT_MATRIX_INPUT**).

13. Fixed bug in computation of hourglass force for pentahedron elements.

14. Fix for ***PART_STACKED_ELEMENTS**: memory overwrite. Models could fail with "Error 10183 (KEY+183), part ... not defined", even if that part did not exist at all.

§7.12 EM (Electromagnetic Solver)

1. Fixed current output in `em_circuit.dat` for Resistive heating solver (***EM_CONTROL**).

2. Fixed issue which made the Eddy current solve (***EM_CONTROL**) slower.

3. Fixed bug in computation of nodal Joule heating rate for shells in MPP

4. Fixed bug to avoid issues when we go from having contact to not in Eddy Current and Induction Heating cases.

5. Fixed issue with computation of the (two-form) current density vector, force, and Joule heating rate with orthotropic materials

§7.13 Fatigue Cards

1. Fixed bug for fatigue analysis with the single precision Windows versions. The program crashes without this fix.

2. Fixed a bug in computing fatigue life in time domain fatigue analysis.

3. Fixed a bug in multiaxial fatigue analysis (***FATIGUE_MULTIAXIAL**). The fatigue damage ratio was incorrect (multiaxial correction is not enabled) before this fix.

§7.14 ICFD (Incompressible Fluid Solver)

1. Several memory bug fixes.
2. Fixed re-meshing issue for multi-volume models.
3. Fixed issue with ***ICFD_BOUNDARY_FSI_EXCLUDE**. Solid parts were not being excluded from FSI, only shells.
4. Fixed FSI for structural models with tied contact. The initial displacement use to tie two surfaces together should not be considered as a motion in the structure for FSI. This was causing instabilities in FSI during initialization.
5. Fixed boundary layer meshing (***MESH_BL**). The surface mesh was deforming when boundary layer nodes were clamped.
6. Fixed nodal PID assignment on nodes shared by a wall and an embedded shell (***MESH_EMBEDSHELL**).
7. Fix in the calculation of the Ergun correlation inertial term factor (***ICFD_MODEL_POROUS**). It was using the permeability instead of the porosity.

§7.15 Implicit (Mechanical) Solver

1. Fixed bug to make 2D shell type 13 with NIP = 4 work correctly with ***MAT_ADD_THERMAL_EXPANSION** for implicit.
2. Fixed incorrect handling of multiple right-hand-sides during the forward / back solvers for the Power Method. The Power Method extracts buckling modes (see ***CONTROL_IMPLICIT_BUCKLING**. It was broken in R11.
3. Corrected the processing of energy and other such statistics in file `glstat` for the multistep linear solution method (NSOLVR = -1 on ***CONTROL_IMPLICIT_SOLUTION**). Prior to the bug fix, the statistics were additive, meaning the statistics for the new time step were added to the ones from the previous time step. Now the statistics are reset at the beginning of the new time step, so they only reflect the current time step.
4. For ***CONTROL_IMPLICIT_EIGENVALUE**,

- Remove implicit input / output files at the end of any eigencomputation. These are scratch files that are of no interest to the user.
- Corrected an error associated with the the nonsymmetric eigensolver which impacted any problem with damping.

5. Fixed the generation of superelements using ***CONTROL_IMPLICIT_MODES** in MPP when nodes involved were shared across processors.

6. Correct how the load curves for IAUTO and IMASS were being processed for implicit keywords ***CONTROL_IMPLICIT_DYNAMICS** and ***CONTROL_IMPLICIT_AUTO** with the options DYN or SPR enabled. Before if the curves IDs did not match the input order (such as an ID of 97125 for the curve input), then LS-DYNA would given an error. This is not how curves are normally processed. Now they are being processed like all other curves in LS-DYNA where the curve ID and order of input are independent.

7. Fixed application of implicit constraint modes (***CONTROL_IMPLICIT_MODES**) in MPP for the case where a constraint mode is also a shared mode.

8. For ***CONTROL_IMPLICIT_GENERAL**, adjust logic of when the last intermittent eigenvalue computation is performed during an explicit simulation if the last one is performed near the termination time. For instance, if the termination time is 10.00 and the explicit time step is 0.01, the the explicit calculation terminates between 9.99 and 10.00. Therefore, if you wanted an eigenvalue computation at 10.00, it would not happen because the calculation would have already terminated. This change causes the intermittent eigenvalue computation to occur before the termination time.

9. During MPP decomposition, the nodes on an interface (see ***INTERFACE**) may be shared by multiple processes. We fixed a bug in implicit where we were double counting the interface forces for the shared nodes.

10. Correct a long standing problem associated with computing the LaGrange Multipliers for constraint processing in Implicit Mechanics. The displacements on the independent dofs were inappropriately being zeroed during this computation leading to inaccurate reporting of resultant forces for Implicit Mechanics.

11. In the previous releases, we had reduced the size of d3eigv database for models with no rotational dofs like solid element only models. Modal dynamics (***CONTROL_IMPLICIT_MODAL_DYNAMICS**) was still trying to read the nonexistent data for rotational dofs and became lost. This revision corrects that oversight.

12. Prior to this revision Implicit Linear Analysis used the forces at the end of the linear step to compute resultant forces. This did not match expectations of the users. We changed the computation of resultant forces, such as those in bndout, to use the force at the beginning of the linear step.

13. Ordering reuse logic for the MPP implementation of implicit mechanics was improperly removed. This revision restored that error. This is an important time saving feature for very large models with multiple time steps.

14. Fixed issue with implicit explicit switching. The time step was not set properly during the switch and caused problems.

15. Fixed bug in computing numerically dissipated internal energies in implicit. It is currently stored as eroded kinetic energy but will change with R14.

16. Fixed numerical sensitivity issue for solid and shell elements when IACC = 1 on ***CONTROL_ACCURACY**. This applies to implicit only. There was numerical noise for no deformation.

17. Fix bug when implicit is used with LMF = 1 on ***CONTROL_RIGID**. The outputs were wrong.

18. Fixed application of ***LOAD_SEGMENT_FILE** and ***LOAD_THERMAL_D3PLOT** in implicit when time step is cut.

§7.16 Initial Cards

1. For ***INITIAL_STRESS_SECTION**, fixed segmentation fault for ISTIFF < 0 when running SMP on many cores.

2. Fixed bug causing spurious error message STR+675 (“undefined node #”) when using ***INITIAL_VELOCITY_GENERATION** with PHASE = 1 and NX = -999. This affects MPP only.

3. Fixed bug in which ***INITIAL_VELOCITY_GENERATION** yields incorrect velocity for ***CONSTRAINED_NODAL_RIGID_BODY_INERTIA** when ICID is not zero.

4. For ***INITIAL_HISTORY_NODE(_SET)**, single precision executables for Windows or IBM linux may have terminated unexpectedly.

5. Use sufficient amount of memory for ***INITIAL_STRESS_SHELL** buffers. It is needed especially when high order shells are used.

6. Fix for ***INITIAL_AXIAL_FORCE_BEAM** with beam element 1 (Hughes-Liu beam). Available since R12.0, this feature could have lead to various side effects, such as severe beam deformations.

§7.17 Isogeometric Analysis (IGA)

1. Fixed a bug when consistency flag is on (CONST in ***CONTROL_-PARALLEL**) and contact is involved for IGA elements.

§7.18 Load Cards

1. Fixed issue that could cause a segmentation fault when using ***LOAD_MOVING_PRESSURE**.

§7.19 Materials and Equations-of-State

1. Fixed delicate interdependency of arrays in ***MAT_ADD_DAMAGE_-DIEM**. It could have resulted in a memory clobber for certain inputs.

2. Fix for combination of ***MAT_ADD_EROSION** and ***MAT_ADD_-DAMAGE_DIEM/GISSMO**. Values from ***MAT_ADD_EROSION**'s optional cards 4 and 5 could have been ignored.

3. Fix for ***MAT_ADD_DAMAGE_GISSMO**: LCSRS being a table and using logarithmic strain rates (first value negative) did not work correctly before.

4. Fix for ***MAT_ADD_DAMAGE_GISSMO** with HISVN = -(ND+19). It did not work correctly for shell elements.

5. Fix for ***MAT_ADD_DAMAGE_GISSMO** with LCSDG < 0 to avoid incorrect element deletion (only SMP/Hybrid with ncpu > 1).

6. Fix for ***MAT_ADD_DAMAGE_GISSMO** with LCSDG being a 3D table (***DEFINE_TABLE_3D**). The first value in the sub-table was interpreted as logarithmic strain rate, and logarithmic interpolation was used in that case. But it is the Lode parameter, and interpolation should be linear.
7. Correction for ***MAT_ADD_DAMAGE_GISSMO** and ***MAT_ADD_-GENERALIZED_DAMAGE** in full deck restart. Element size for LCREGD and FADEXP < 0 was re-computed after restart, but it should not.
8. Fix for ***MAT_ADD_GENERALIZED_DAMAGE** with `long=s`. The long input format was not working with this keyword before.
9. Fixed bug in ***MAT_ADD_THERMAL_EXPANSION** with solid element formulation 13. Incorrect elastic strain was used resulting in incorrect stress.
10. Fixed ***MAT_NONLOCAL** when used with eroding elements. The smoothing calculation was using uninitiialized data due to a failure to set initial values for the eroded elements prior to smoothing. This error could be subtle or cause an error termination.
11. Fixed ***MAT_ELASTIC_FLUID** output to `d3hsp`. The bulk modulus, k, was described incorrectly as “e” in `d3hsp`.
12. Fix load size inconsistency in linear implicit of ***MAT_ORTHOTROPIC_ELASTIC**. The response was not linear.
13. Fixed bug in computing yield for rate dependent ***MAT_PIECEWISE_LINEAR_PLASTICITY**. Sometimes (rarely) the interpolation between load curves failed.
14. Fix for combination of ***MAT_024** (shells) with TDEL > 0 and GISSMO. History variables conflict could lead to wrong results.
15. Fixed the use of material types 26, 126, or 201 with thick shell elements when AOPT = 2 or 3. The material directions were being miscalculated causing wrong stress.
16. For ***MAT_030**:
    - Fixed plastic convergence issues when using LCSS.
    - Fixed wrong load curve ID for LCSS.

- Fixed issue that could cause a segmentation fault.

17. Fixed bug that occurs when ***DEFINE_MATERIAL_HISTORIES** is used with ***MAT_FABRIC**. History data was being clobbered.

18. Fixed outdated argument lists for user subroutines for ***MAT_-USER_DEFINED_MATERIAL_MODELS** / ***MAT_041-050**. Before this fix user some defined features (`usrshl`, `usrsld`, `usrfrc`, `usrtie`, `usrtbrk`, `usr_nunonl`, `mortar_*`, `rebar_*`) using ***MOODULE** did not work correctly.

19. Fixed bug that occurs when ***DEFINE_MATERIAL_HISTORIES** is used with ***MAT_USER_DEFINED_MATERIAL_MODELS**. The bug led to an inconsistent count of history variables.

20. For ***MAT_053**, extrapolate load curve for von Mises stress as a function of negative volumetric strain using the first 2 points of the load curve if the volumetric strain is less than the first point of the curve. This fix prevents an error termination.

21. For ***MAT_054** (***MAT_ENHANCED_COMPOSITE_DAMAGE**):

    - Set EC = EB, if EC is undefined. If EC was undefined, LS-DYNA might have stopped unexpectedly.
    - Set upper bound for strengths (XC,...=1.e+16) to avoid possible floating overflow in single precision.
    - Fixed initialization of material angles. Before, the simulation could error terminate due to out-of-range forces when ***AIRBAG** definitions were present in the input.

22. For ***MAT_058** (***MAT_LAMINATED_COMPOSITE_FABRIC**):

    - Fixed bug for LCDFAIL check. Prior to this fix the LCDFAIL option did not work for shells.
    - Fixed bug for using $\geq$ 8 digit load-curve ID for GAB.
    - Fixed contact stiffness in case EA, EB, or GAB are defined with load curves or tables.
    - Added a check if load-curve /table is defined for a proper strain range and issue an error message for shells and EA or EB < 0.

23. Fix for ***MAT_076** with temperature shift. Initial temperature in an uncoupled analysis is not always zero (such as that coming from ***LOAD_THERMAL**).

24. Fix for combination of ***MAT_077** / ***MAT_181** with VFLAG = 1 and ***INCLUDE_TRANSFORM**. Normalized shear relaxation moduli Gi (unitless) should not be converted.

25. For ***MAT_077**, fixed bug for shell elements that could cause instability.

26. Fixed incorrect effect of BVFLAG in Card 2 field 1 of ***MAT_FU_CHANG_FOAM** / ***MAT_083** on initial time step size.

27. ***MAT_WINFRITH_CONCRETE_REINFORCEMENT** / ***MAT_084_REINF** did not work in MPP when the reinforced elements were defined with EID1 and EID2 (Card 1a). It worked correctly when defined by the PID method (Card 1b).

28. Fixed bug in ***MAT_100** (***MAT_SPOTWELD**) for OPT = 0/-1 when using ***DEFINE_CURVE**s for defining force / moment resultants as a function of the effective strain rate, by setting NRR, NRS, NRT, MRR, MSS, MTT as negative values. Added a more meaningful error message in case load curve is not defined.

29. For ***MAT_SPOTWELD_DAMAGE-FAILURE** with beams, fixed bug that could have led to earlier damage evolution than anticipated.

30. Fix initial temperature increment in ***MAT_106** for shell elements. In a structural-only analysis, the ″old″ temperature should be set to the initial temperature, such as from ***LOAD_THERMAL_LOAD_CURVE**.

31. Fixed incorrect computation of phel for ***MAT_110** / ***MAT_JOHNSON_HOLMQUIST_CERAMICS** when you iput phel = 0 in the data cards.

32. Fixed spurious solid element deletion for ***MAT_110** / ***MAT_JOHNSON_HOLMQUIST_CERAMICS** at $t = 0.0$.

33. For ***MAT_122_3D**, updated plasticity algorithm due to implicit problems.

34. Fix AOPT < 0 for ***MAT_122_3D** and also add missing `d3hsp` echo of material parameters.

35. Added support for ***MAT_MODIFIED_HONEYCOMB** / *MAT_126 with solid element types 0 and 9 to work with ***CONSTRAINED_TIED_NODES_FAILURE**.

36. Fix combination of ***MAT_126** and ***MAT_ADD_THERMAL_EXPANSION**. It only worked for ELFORM = 0, but now also works for others.

37. Rotate back strains for ***MAT_136**. Bug resulted in wrong strain output for post-processing.

38. Change description of ***MAT_142** from ′anisotropic crushable foam′ to ′transversly isotropic crushable foam′ in `d3hsp` to be consistent with the Users′ Material Manual.

39. ***MAT_155** will now call subroutine `matusr_24` for FS < 0.

40. For ***MAT_157** (***MAT_ANISOTROPIC_ELASTIC_PLASTIC**):

    - Fixed implicit convergence problem due to division by zero.
    - Fixed wrong tangent stiffness.
    - Corrected the update of transveral (through the thickness) strain. Prior to this correction, the thickness strain was not computed correctly and did not fullfil the condition that was set with the Lankford parameters (R-values). This applies to shells only.

41. Implemented viscoelastic option for ***MAT_SIMPLIFIED_RUBBER/-FOAM / *MAT_181** for solid formulation 13, tetrahedron with nodal pressure.

42. Fix for keyword input read of ***MAT_186**: variable TSLC2 was not working in free format or with ***PARAMETER**.

43. Resolve single precision issue with small BETA values in ***MAT_187** to increase accuracy of results if LCEMOD > 0.

44. Fix problem with `long=s` for ***MAT_196** to avoid error termination.

45. ***MAT_208** (***MAT_BOLT_BEAM**) bug affected R12.0 only. When AXSHFL = 0 and LCSHR was a ***DEFINE_TABLE**, the shear force did not correctly follow the relation with displacement defined in the ***DEFINE_TABLE**. It worked correctly when LCSHR was a ***DEFINE_CURVE**, and it worked correctly in R11 with both ***DEFINE_-CURVE** and ***DEFINE_TABLE**.

46. Make option E < 0 of ***MAT_224_GYS** work correctly, which was not the case before.

47. Fixes for ***MAT_PIECEWISE_LINEAR_PLASTIC_THERMAL** (***MAT_255**):

- Avoid error termination if huge number of curves (> 9999) is used in a model.
- Avoid error termination with large curve IDs for E < 0 and PR < 0.
- Fixed temperature-stress coupling for shells. Before fix a thermal gradient in thickness would not affect element stresses.

48. Fixes for ***MAT_258**: repair visco-plasticity, use correct thickness of shell type 16, and eliminate potential dependence on ncpu.

49. Fix bug for ***MAT_264**/ ***MAT_TABULATED_JOHNSON_COOK_ORTHO_PLASTICITY** which gives incorrect results when running with adaptivity.

50. Improve numerical accuracy for ***MAT_BERGSTROM_BOYCE_RUBBER**.

51. Suppress rate effects in ***MAT_BERGSTROM_BOYCE_RUBBER** according to the rules of IRATE on ***CONTROL_IMPLICIT_DYNAMICS**.

52. For ***MAT_274**, fixed bug where shell thickness strain was not updated properly.

53. Fixed issue that could cause a segmentation fault when using ***MAT_278** / ***MAT_CF_MICROMECHANICS**.

54. Fixed the use of equation of state materials with thick shells, when ***PART_COMPOSITE** or ***INTEGRATION_SHELL** was not used to define an integration rule.

§7.20 MPP

1. Add MPP support for ***TERMINATION_BODY**. It was never properly implemented in MPP.

2. Fix for MPP handling of ***DEFINE_CURVE_FUNCTION**. Specifically, if the built-in function PIDCTL was used, along with MPP predecomposition, the behavior was unpredictable due to a memory clobber.

3. Fixed bug that caused MPP to improperly handle a `d3kil` file if a jobid is being used. The result was that it would ignore the contents of the file and always terminate.

4. Fixed bug that resulted in MPP reporting incorrect rotational masses to `massout`.

§7.21 Output

1. For ***DATABASE_CURVOUT**, fix thermal-only solutions to write the `curvout` file in LSDA format. The ASCII version was working, but if the LSDA format was requested, then this data was not being written at all.

2. Fixed `d3plot` file corruption when eigenvalues are computed.

3. For linear DK shell elements (***SECTION_SHELL**, ELFORM=18), the moments are now correctly written to the cross section output generated by ***DATABASE_SECFORC**. Before, incorrect values were reported.

4. Number of parts reported in `d3eigv` is wrong for the SMP version.

5. Restored how encrypted files are echoed to `d3hsp` back to how it was done in version R9.3.1.

6. For ***DATABASE_RECOVER_NODE**:

   - Issue warning message when no vertex is detected for the Zienkiewicz-Zhu method (METHOD = 0).
   - Take care of special case where Zienkiewicz-Zhu method is applied to a mesh without any vertex. Previously it could causes a segmentation fault in some cases.

7. For ***DATABASE_CROSS_SECTION**, fixed a bug for cross sections that contain 2D seat belts.

8. Fixed the reported mass to `d3plot` when tet form 10 and tet form 13 erode. Eroded mass was not being subtracted.

9. Fixed bug causing NaN in `secforc` when using beam type 6 with zero area and length.

10. If NEIPH/NEIPS in Card 1 of *DATABASE_EXTENT_BINARY is negative, reset it to positive and issue warning message, KEY+1396.

11. Fixed incorrect legend of NODOUT in `binout` for SMP.

12. Output forces/moments to `secforc` for beam type 13.

13. Output strain tensor to `d3part` when STRFLG > 0 in ***DATABASE_EXTENT_D3PART** and STRFLG = 0 in ***DATABASE_EXTENT_BINARY**.

14. Fixed zero torsion output to `swforc` for beam spot welds.

15. Fixed bug that caused zero damping energy in `GLSTAT` when using ***DAMPING_RELATIVE**.

16. Fixed bug that caused wrong mass output in `SSSTAT` (***DATABASE_SSSTAT_MASS_PROPERTIES**).

17. Fixed bug that caused wrong kinetic energy output in `GLSTAT` when using ***INITIAL_VELOCITY_GENERATION**

18. Fixed bug that caused wrong output for ***DATABASE_NCFORC_FILTER** in MPP.

19. Fixed a bug in the moment computation for the moments output to `rcforc` (***DATABASE_RCFORC_MOMENT**). This bug only affects post-processing, not the analysis itself.

    bug fixes

20. When NINTSLD was set to 8 (see ***DATABASE_EXTENT_BINARY**), the `d3thdt` output file was corrupted. Results could still be obtained via the `ascii` and `binout` files, only the `d3thdt` file was affected.

21. Occasionally an analysis could terminate abnormally when beam elements were included in ***DATABASE_CROSS_SECTION** and one or more of the elements eroded.

22. Fixed possible bug in output of stresses for anisotropic materials when using CMPFLG = 1 on ***DATABASE_EXTENT_BINARY** and ***DEFINE_MATERIAL_HISTORIES**. Prior to this fix, the output stresses might not have been properly transformed into the material frame. This only affects the post-processing and not the analysis itself.

23. Fix output to `d3hsp` of ***DEFINE_MATERIAL_HISTORIES**. An incorrect number of variables and associated labels were reported.

24. Fixed output of effective plastic strain for ***DEFINE_MATERIAL_HISTORIES**. The values were incorrect.

25. Fixed corrupted `d3eigv` when PENOUT or ENGOUT are used on ***CONTROL_OUTPUT**.

26. Fix missing support of options LCUR and IOOPT for ***DATABASE_-DCFAIL**.

27. Add missing legend / title to ***DATABASE_DISBOUT** output files.

§7.22 Restarts

1. Two fixes for ***DATABASE_BINARY_INTFOR** and full deck restart:

   - SMP was opening the new file with the wrong "max file size" during restart, causing problems reading the data in some cases
   - MPP was opening the new file without the "ab" family extension, and overwriting the files from the initial run.

2. During a simple restart, only open an ***INTERFACE_LINKING** file on processor 0 in MPP. The other processors were trying to open the file with a nonsense name, causing a non-obvious "file open" error message.

3. Bug that caused LS-DYNA to ignore the output interval specified in the restart file for RCFORC during a small restart.

   Fixed error for *DEFINE_MATERIAL_HISTORIES and DEFINE_-FUNCTION_TABULATED for newformat=long option

4. Generic restart, memory bug that could affect all double precision restart jobs. This bug affects all executibles since July of 2012.

5. Fix bug associated with a full deck restart for ***CONTACT_-TIEBREAK_SURFACE_TO_SURFACE**. This affects SMP only.

6. Fixed seg fault with full deck restart when initial run is 6 degrees-of-freedom while the full deck restart run is 3 degrees-of-freedom.

§7.23 Sensors

1. Fixed an MPP bug for ***SENSOR_DEFINE_FORCE** with TYPE = JOINTSTIF. It didn't trace the force correctly when more than one core is used.

2. Fixed a bug for ***SENSOR_DEFINE_ELEMENT** with TYPE = STRAIN. It occurred when STRFLG of *DATABASE_EXTENT_BINARY is 0.

3. For ***SENSOR_CONTROL**:

   - Fixed a bug for TYPE = ELESET when the element set type is discrete. It failed to delete initially off discrete elements.
   - Fixed bug for TYPE = CNRB that caused failed constrained nodal rigid bodies to not be deleted.
   - Fixed bug for TYPE = SPC that occurred when trying to turn on SPC in the middle of an analysis.

4. Fix for ***SENSOR_DEFINE_FORCE** with CRD $\neq 0$ and `long=s`. Error in structured read happened (STR+211) with MPP and np > 1.

§7.24 Thermal Solver

1. ***DATABASE_EXTENT_BINARY** now honors IEVERP in thermal only calculations.

2. ***BOUNDARY_RADIATION_SET** (including valid keyword options) now acknowledges default settings.

3. Fixed bug in `d3plot` output of the thermal solver. For ***CONSTRAINED_NODAL_RIGID_BODY** cards in a thermal-only analysis, the `d3plot` files were corrupted and could not be read by LS-PrePost. The solution found by the solver was correct.

4. Fixed some bugs with constraint handling in thermal solver in order to avoid redundant constraint errors in LCPACK:

   - Fix of ***CONSTRAINED_ADAPTIVE** defined on solid/beam nodes when thermal thick shells are also present in the model.
   - Combination of temperature boundary conditions and other constraints on the same node. DoFs constrained by ***CONSTRAINED_ADAPTIVITY** and ***CONSTRAINED_SPOTWELD** are excluded from the ***BOUNDARY_TEMPERATURE** constraint.

5. Fixed a memory bug that could cause strange bahavior when the PSEROD flag was active for ***BOUNDARY_FLUX** condition.

6. Bug fix for variable time step control in thermal solver in MPP (NP > 1). Bug may have caused the simulation to hang.

7. Fixed heat density distribution for stationary heat source in ***BOUNDARY_THERMAL_WELD**.

8. Fixed a bug in the `binout` output of thermal solver for MPP. May have caused inconsistent data in the tprint section of the `binout` file and segfault when using l2a.

§7.25 XFEM (eXtended Finite Element Method)

1. Fixed a memory bug so 2D XFEM (ELFORM = 52 on ***SECTION_-SHELL_XFEM**) now works for plastic materials.

2. Fixed a bug in ***BOUNDARY_PRECRACK**. Gave warning message when precrack points are changed to avoid the precrack passing through nodal points.

§7.26 Miscellaneous

1. Fix indexing error related to ***SET_SEGMENT_GENERAL**, which could have resulted in incorrect values being applied for the DA1-DA4 parameters, or possibly a segmentation fault during input processing.

2. Corrected some bugs related to ***INTERFACE_LINKING** that could lead to segmentation faults.

3. Store file name and format flag for ***INTERFACE_COMPONENT** into the structured file. With this change, if we convert keyword input to structured input and run from structured input, LS-DYNA will know the required file format for the component data. Previously LS-DYNA always produced a non-LSDA file when running from a structured file.

4. To improve memory performance, removed initial large allocation that was zeroing all of memory and thereby physically allocating it on Linux. This change restores behavior that existed in releases prior to R11.2.

5. Fixed issues with ***DEFINE_FUNCTION** in structured input. Also made sure they work properly with predecomposition.

6. SMP seg faults when a file with a long filename is in the working directory. The error occurs when the length of the filename is at least 188 characters.

Listing 1

```
*PARAMETER
$    PRMR1      VAL1     PRMR2      VAL2
RuPLscXXa1 &R1uPllSARuPLscXXt1       0.0
```

7. Fixed structured input error reading discrete element

8. If the number of valid license days is less than 7, change the error to warning.

9. Fixed bug for ***PARAMETER** if long format is used.

10. Character parameters (see ***PARAMETER**) were not conforming to the usual convention for comma-delimited input.

11. Fixed bug when using a character parameter (see ***PARAMETER**) in a larger string as part of an include file name. The bug caused an extra space in the file name which led to an error termination.

12. Fixed bug in reading data cards of ***PARAMETER** that occurs if there is no space between a symbol and a value like in Listing 1.

13. For ***DEFINE_CURVE_FUNCTION**:

    - Fixed a bug for PIDCTL that occurs for thermal-only analyses. It resulted in an erroneous PID control force.
    - Fixed an MPP bug for FX, FY, FZ, TX, TY, and TZ that might cause incorrect results.
    - Fixed a bug for the function SENSOR that sometimes caused a job to fail with a spurious message.

14. ***MODULE_PATH** did not work correctly for user defined features.

15. For ***DEFINE_PRESSURE_TUBE**:

    - Fixed automatic generation of type 25/26 shells.
    - Fixed bottleneck in MPP when using many rigid parts in the model.

16. Enabled ***DEFINE_MATERIAL_HISTORIES** to be used with Rayleigh damping. The history data and damping variables were conflicting, causing early failure of elements.

17. Fixed a problem with ***NODE_TRANSFORM** when used with spot weld assembly generation. Spot weld elements were getting distorted.

18. Fixed segmentation fault that occurs during the output of a `d3plot` file after a solid element is deleted. This behavior occurred for user material models.

19. Fix 'reference load curve ID # cannot be found' error, KEY+825, when load curves for ***DEFINE_TABLE_2D/3D** are contained in a different include files after the include file that contains the ***DEFINE_TABLE_2D/3D** statement.

20. Fixed bug causing error 'BEAM INT ID ### does not exist' (KEY+308) when using ***INTEGRATION_BEAM** with ***INCLUDE_-TRANSFORM**.

21. Fixed issue with ***DEFORMABLE_TO_RIGID_AUTOMATIC** when CODE = 4 and OFFSET > 0.0. Offset was not working correctly.

22. Fixed incorrect penetration value in warning message, INI+226, for ***RIGIDWALL_PLANAR**.