Oasys Ltd

LS-DYNA Environment Version 11

Update and Release Notes

Oasys Ltd

The software house of Arup

Contents

1 Introduction

The Oasys Ltd LS-DYNA Environment 11.0 suite, dated April 2013 is a full release which supersedes all 10.2 releases.

This document lists the enhancements that have been made during development, the bugs that have been fixed (since 10.2). Each "case id" referred to in these lists is unique and should be cited if more information or clarification are required.

1.1 Compatibility with LS-DYNA releases

The 11.0 release has been used and tested with LS-DYNA version R7.0, and supports all the keywords in volumes I and II of the R7.0 User's Manual dated January 2013 (except *EF). The keywords in volume III (Multi-Physics solvers) are not supported.

LS971 R6.0 is now the default keyword output format from PRIMER. R6.1 and R7 are also available and can be set as preferences.

1.2 Hardware platforms supported

The complete Oasys Ltd LS-DYNA Environment 11.0 is available on the following platforms:

Windows	XP, Vista, Windows 7	32 bit
	XP 64, Vista 64, Windows 7 64bit	32 and 64 bit
Linux 2.6 kernel	Redhat Enterprise 4.0	32 and 64 bit

We build Linux 2.6 kernel versions on both SuSE and RedHat Enterprise Linux, however we have found that the executables seem to be completely interchangeable with no problems or loss of performance.

For simplicity of support, maintenance and testing only the RedHat Enterprise 4.0 build has been released. If you require a different Linux build, please contact us.

1.3 Graphics hardware supported

The default graphics protocol used on all platforms in the table above is OpenGL, which gives good quality 3D performance on a wide range of hardware.

"Legacy" X-Windows (2D only) graphics drivers remain in PRIMER and D3PLOT on Unix and Linux platforms since these are used by a small minority of clients who display over a network of mixed machine types.

The 2D X-Windows driver has been withdrawn from T/HIS for technical reasons, however it is now linked with both hardware OpenGL (the default) and the MESA emulator on UNIX and Linux. MESA emulates OpenGL in software on X-Windows; therefore clients using a mixed machine environment with only 2D X-Windows support will still be able to display graphics.

1.4　　　FLEXlm Licensing

The version 11.0 software has been compiled using version 11.11 of FLEXlm, while the previous version 10.x software used version11.8 of FLEXlm.

The version 11.0 licence file and licence daemons are backwards compatible with both the 10.x and 9.x software releases. This means that any existing 10.x and 9.x software will continue to work with the new licence files and servers.

We recommend that you update all of your Oasys Ltd LS-DYNA environment license servers with the files shipped with the version 11.0 software.

2 Bugs Fixed

The enhancements and bug-fixes for each program have been broken down into a series of topics. Within each topic enhancements and bug-fixes are listed by case ID (most recent first).

2.1 PRIMER

2.1.1 Bugs Fixed in 11

Airbag

- **Case 21011**

In some cases the *AIRBAG editor could 'lose' parameter associations in data fields. This could occur in cases where the keyword's contents and number of rows depended on the details of its entries. In this example the problem showed up in the case of a WANG_NEFSKE definition in which the field POP determines whether or not some rows of data are present. In order to deal with this the airbag editor had to reformat itself, transferring 'old' data to 'new', and it was losing parameter associations during this transfer. Now fixed.

- **Case 20530**

The units were incorrect for the LOU field of *AIRBAG_SIMPLE_AIRBAG_MODEL. This has been corrected.

- **Case 20499**

Two separate but related problems to do with *AIRBAG_PARTICLE are solved in this case.
(1) An example deck had a data field '30.00' which was misaligned so that the last trailing zero spilled over into the next data field, thus (E10.0, I10 format):

```
$2345678901234567890
        30.00         1
```

PRIMER (correctly) treated this as an error since the 2nd integer field contained both zero (0) and one (1) with spaces between them, meaning that the input deck was not read. However LS-DYNA read this input deck quite happily.
Therefore PRIMER now detects this 'zero spillover' problem, and while it still treats this as an error by default, in doing so it prints a message suggesting that the user try setting the 'zero spillover' flag on the Read Options panel.
Setting this flag treats this as a warning, not an error, ignoring initial zeros followed by spaces in an otherwise valid field, thus reading the deck.
This is not the default because it is potentially dangerous. In the example above truncating a trailing zero after the decimal point to turn '30.00' into '30.0' is harmless, as doing so does not change its value. However consider the case of an integer '300|0' truncated to '300' ... its value has changed.
The correct solution is to investigate and fix the problem manually.

(2) This deck also contained a *DATABASE_ABSTAT_CPM card, which

PRIMER rejected because it is not in any keyword manual. LS-DYNA treats this card as identical to *DATABASE_ABSTAT, and Primer now does the same.

- **Case 20465**

If airbags are created from other keywords (daisy chained popup) then the optional label is on by default. Previously it was off which meant when creating, the calling panel was not updated with the new airbag.

- **Case 20448**

Nodes on airbag shell reference geometry as well as the 'normal' shell nodes were merged during node replace/merge. An option has been added to be able to control this.

- **Case 20400**

It has historically been a problem that parameterised data fields in *AIRBAG definitions could get lost (reverting to their underlying values) in the airbag editor if some change was made that altered the number or alignment of rows. Typical changes might be adding or removing the _TITLE definition, but some airbag types can alter the number of rows if certain data fields change.
This has now been fixed: parameters should persist through all editing processes, and they should also be 'remembered' if the airbag type is changed and data is moved from old to new data fields with the same names.

Airbag folding

- **Case 20355**

The graphics colours in the airbag folder were not correct when using a white background. Now fixed.

- **Case 18369**

Null folds can now be written from the airbag folder.

Assembly

- **Case 21447**

PRIMER could hang when trying to remake a connection defined by an assembly. Now fixed.

Attached

- **Case 19843**

The model class JavaScript functions UnblankFlagged() and BlankFlagged() did not work correctlywith mixed element and part flagging (only the parts were blanked/unblanked, not any extra flagged elements). This has been corrected.

Belts

- **Cases 21899 & 21960**

If a seatbelt is attached to structure 'indirectly' via something that uses a node set, for example by

+ A nodal rigid body
+ A rigid node set
+ An extra node on a rigid body defined by a node set rather than an explicit node

Then if the node used is not attached to anything other than a belt element it will get deleted in the 'delete old belt' phase prior to manual remeshing. This is because of PRIMER's default deletion logic that permits nodes to be removed from sets, and the consequence is that the belt loses its connection to the structure at that point.

Special logic has been added so that nodes meeting the criteria above are no longer deleted in the pre-meshing deletion phase if they are used as the belt meshing segment end points.

(Note: this problem did not arise during auto-refit since it uses a different deletion sequence).

- **Case 21959**

The seatbelt fitter allows explicit 'twist' to be defined at path points, and for fixed points the twist about 2 axes may be defined.

This twist was being honoured for points with a slipring or retractor defined, but not at fixed end points with no such elements.

Now fixed: explicit twist is retained at fixed intermediate and end points whether or not these points have a retractor / slipring defined.

- **Case 20461**

The seatbelt fitter would crash during the meshing phase if shells were being used in the belt and the SECTION_SHELL card was 'Latent' (meaning referenced but not defined).
However fitting and/or meshing a seatbelt with a latent section definition for the elements is not recommended as the thickness of the belt elements will be undefined, and this may give rise to initial penetrations if the thickness ultimately used is greater than that used in the fitter.

- **Case 20109**

When fitting seatbelts the basic path of the belt may have points defined by coordinates in space, or by nodes. In the latter case the nodal coordinates superseded the coordinates on input, replacing the explicit coordinates with the current nodel coordinates if different, but this check was not performed subsequently.

Therefore if a nodal coordinate used for a path point was changed interactively during the PRIMER session the belt fitter did not 'know' about this and would ignore it, using the belt path coordinate as originally defined.
This has been corrected: prior to any belt fitting operation all path points defined by nodes have their coordinates updated to be those of the node.

- **Case 20097**

When refitting a 2D seatbelt manually, and reusing existing 2D sliprings and retractors, the fitter might 'forget' the rigid bodies to which these were attached.
As a consequence the node sets used to locate them would no longer be part of the relevant rigid body, and instead would become free-standing nodal rigid bodies.
This has now been corrected: a parent rigid body to which these items are attached is detected prior to any deletion that may take place, and is 'remembered' for use in subsequent meshing if they are reused.

- **Case 20094**

When defining a belt path users normally click on nodes, and these nodes plus their coordinates are stored in the 'path' definition for future fitting, which all worked well.
However in special cases where path points are tweaked manually to move them away from node locations a problem was arising. Part of the belt fitting algorithm is to look for the nearest nodes on dummy or structure, as these can be used for a variety of purposes. For example if the dummy is moved then a refitting operation knows that it must take into account the movement of these nodes when projecting the belt forwards prior to refitting.
However as part of this process the 'nearest node' was made to be the 'path node' at any point where no explicit node was defined, with the result that in subsequent fitting operations the belt path would be moved - usually in an unwanted direction.
This is now fixed. The nearest nodes are still computed and stored internally as before, but they are no longer substituted into the belt part.

- **Case 19537**

The seatbelt fitter crashed when attempting to fit to a dummy if some of the dummy elements used as dummy 'structure' for fitting were 6 noded 'wedge' elements.
The workaround was to fit to dummy shells only (which was also much faster).
This has been fixed in V11.

- **Case 18894**

The seatbelt fitter in PRIMER has been criticised for being slow when a belt contains many elements across its width, or the 'structure' model is complex.
This is because a lot of geometrical calculations must be performed, and also a lot of contact between belt and structure computed. The latter, in particular, rises in complexity as the product of belt mesh density times structure mesh density.
The speed of the belt fitter has been improved in two ways:

(1) The internal storage of geometry has been revised to remove a significant amount of 'double handling' of data. This gives an improvement in speed of around 20%.
(2) Much more significantly the contact algorithms have been parallelised, giving

a dramatic improvement in speed.

The combination of these two changes gives a speed-up of between 2 and 3 times on a typical four-core machine.
However parallelising contact has required some re-ordering of the sequence of calculations, and in some models this can result in a change in the belt shape when compared with older versions of PRIMER.
Normally this difference is minimal, but in a few cases a minor change in contact penetration can result in the belt taking a different path with the result that the final shape is significantly different. This is not an error, rather it suggests that the geometry of the problem is somewhat ill-conditioned.
In recognition of the fact that different results from later versions of software can cause problems there is a switch to turn off parallelised fitting, causing reversion to the linear pre-V11 behaviour. This can be controlled interactively, or made persistent via a preference.

Checking

- **Case 21855**

In rare cases LS-DYNA models refuse to initialize giving non coincident joint node error when PRIMER check has not found the error. To avoid this, the test tolerance has been changed from 1e-3 to 0.99999e-3.

- **Case 21482**

If using CONSTRAINED_NODAL_RIGID_BODY_INERTIA in PRIMER, you could get incorrect error messages written to the dialogue box when running a model check. This has been corrected.

- **Case 21405**

You could see an incorrect error message when checking a contact about no entities on the master side of the contact if the slave side is a shell set. This has been corrected.

- **Case 21352**

The part added mass/percent added mass model check, incorrectly reported an error for parts of element type for which PRIMER cannot calculate the timestep (e.g.2d seatbelt elements).

- **Case 20617**

PRIMER could crash when checking models that contained *DEFINE_TABLE and *DEFINE_FUNCTION definitions. This has been corrected.

- **Case 20357**

Missing checks for *CONTACT_2D have been added.

- **Case 19937**

*MAT_NONLOCAL definitions must be sequential in label. In some cases, the check for this in PRIMER could be skipped. This has been corrected.

- **Case 19791**

Added Error check exceptions for BEAM_056 to exempt MAT_206, 209.
Also, added exception for mat type 209 in error check PART_171.

- **Case 19766**

MAT_209 => IAX, IHARD, ALPHA, BETA, GAMMA all have sensible defaults in LS-DYNA. It is not an error if any of them are zero. Corrected the error check.

- **Case 19318**

The logic which checks for inconsistencies in Material restraint, Boundary applied motion and initial velocity for rigid bodies has been improved.

Clipboard

- **Case 21725**

PRIMER could crash if using the clipboard with a model which has a long title (80 characters). This has been corrected.

- **Case 20516**

When items are loading into the clipboard a 'clipboard' row appears in object menus if one or more items of the relevant type are in the clipboard, and the clipboard contents can be selected for that operation.
When filtering is applied, for example limiting shells to those of a given thickness, the '#items in clipboard' total is updated to show only those items in the clipboard that pass the filter, and selection will only pick those items. This is correct behaviour.
However if the default 'sketch when you hover over' behaviour was active it would show all items of the given type in the clipboard, ignoring any filter settings. In the example above it would show shells of any thickness. This was the function that sketched the clipboard contents was not considering the current filter settings.
This has now been fixed: hovering over the clipboard row in an object menu will now only sketch items that are in the clipboard AND which have not been excluded by any active filters.

Coating

- **Case 21012**

When picking by category 'element', which can happen explicitly in contexts such a Blank and Quick Pick, but also implicitly in contexts such as Coat Part as cited here, picking might sometimes fail to select an element.
The reason was that 'element' picking loops over all element types, and now that picking can be parallelised if there are enough elements in a category to warrant it

there is the possibility of ending up with a mixture of 'elements picked in parallel' and 'elements picked using scalar logic'.

There was an error in the mixing of parallel and scalar picked elements which could result in those picked by the scalar method getting 'lost'. This would be most likely to occur when picking solid elements from a model with relatively few solids but many shells. That was the problem in this case: Coat Part implicitly picks solids, and these were getting lost.

Now fixed.

Connections

- **Case 20954**

When changing an assembly name, if a connection layer was defined by an assembly the connection was not updated. This has been corrected.

- **Case 20953**

The 'More' panel opened from the connnection table layer popup was very small when not in 'multiple' mode, so opening a selection menu was too small to be useful without expanding. The panel is now larger when mapped.

- **Case 20576**

A reprojection of connections when multiple models are read into PRIMER could result in the part ID referenced by the connection entity being corrupted. This has been corrected.

- **Case 20310**

Consistency for alignment of beams created has been improved for mixed case bolts which do not have hole on every layer. The beam will be normal to the hole in preference to the projected shell.

- **Case 20237**

If a connection joined one part to itself, you would have to turn on the 'same part' setting to remake successfully. This is now done automatically upon remake.

- **Case 20057**

When deleting a part, even if you had NO_ACTION set for the connection action, the contents (FE) within any attached connections would appear in the list of entities to be deleted. The contents would not be deleted, but they should not appear in the list for this setting. This has been corrected.

- **Case 19579**

Model check on model with connections was giving error messages about unsupported type 2147483647. This has been fixed.

- **Case 19572**

When the connection tool is pressed, if the connection entity switch is off it is turned on and all connections become visible. We have changed this so that only

the connections associated with unblanked elements or parts are now made visible.

- **Case 19571**

Part replace sometimes changed the status and colour of connections remote from the actual parts being replaced. This has been fixed.

- **Case 19487**

When creating connections from geometry points/curves the selection menu for the geometry data would be mapped on top of other buttons. This has been corrected.

- **Case 19475**

When trying to connect 3 panels, if the resulting connection would be too long when compared to PRIMER's max/min length settings, PRIMER would refuse to make the connection even if a 2T connection would be OK. This has been modified so the 2T weld would be made by default.

- **Case 19381**

Sometimes 'From FE' for adhesive would not create all the connection entities it could from all the selected solids. This has been corrected.

- **Case 19323**

The 'From FE' function to make bolts used very simplistic logic to determine the diameter and length of the bolt. This has been improved to determine a more realistic cylindrical volume.

- **Case 19105**

If a connection fails to make because the same part is used in multiple layers and the global setting to allow conection between same parts is off, PRIMER will now give a more useful error message.

- **Case 18595**

When deleteing some solids from an adhesive run, all solids within the run would also be flagged for deletion. This has been modified so that only the selected solids are flagged for deletion. This is for the default CONX_ACTION setting of 'Empty conx'.

- **Case 18082**

Spotweld lines sketching was lost when plotting a pdf when you currently have a black background and specify a white background output. This has been corrected.

Constraints

- **Case 21445**

The 'undo' functionality on the CONSTRAINED_SPOTWELD/RIVET/GENERALIZED_WELD panels did not work when creating a non-labelled item when you had include label ranges set for the current include. Now fixed.

Contacts

- **Case 21958**

The algorithm that determines the thickness of segments on solids which are involved in tied contact has been improved to match LS-DYNA's methodology. Users will observe that MPP and SMP methods give very different results with respect to whether or not nodes tie to solid elements.

- **Case 20930**

Rearranged text description on Contact keyword cards A-E to fit the space.

- **Case 20810**

PRIMER could reset fields in the optional cards of contact cards if certain other values were not set. For example, SBOPT was reset to zero if updating a card where the SOFT field was not set to 2. This has been corrected.

- **Case 19947**

The default for field INITITER on *CONTACT_..._MPP was incorrectly treated. In this case a blank entry means use the default value of 2. A zero entry means use value of zero. This has been corrected in PRIMER11 and additionally, any decks written in previous versions of PRIMER (or other software) where zero is read for this entry will incur a warning message and possible correction (at the user's discretion) on keyin.

- **Case 19423**

When node 3 of a beam on the slave side of a contact belonged to a segment on the master side, PRIMER incorrectly reported an error. Now fixed.

- **Case 19392**

MPP LS-DYNA appears to ignore contact parameters when determining which nodes are tied. The PRIMER contact checker is in MPP mode by default and will also ignore this setting. The preference primer*contact_check_mode: SMP may be set for those that do not use MPP.

- **Case 12098**

The logic for determining the correct contact parametric from MAXPAR (SMP) and PARMAX (MPP) has been added. Note that the PRIMER penetration check is now in MPP mode by default but may be configured to SMP by the preference primer*contact_check_mode

- **Case 21613**

The _ONLY option was not supported for
*CONTACT_TIEBREAK_SURFACE_TO_SURFACE. This has been corrected.

Contour

- **Case 21383**

When *PART_INERTIA had rotation velocity defined and used NODEID to set
the centre of gravity, the nodal velocities were not correctly calculated. Now
fixed.

- **Case 21360**

PRIMER was not contouring timestep for seatbelt elements. The missing
calculation has been supplied.

- **Case 21285**

The effect of sorting for triangular, tetrahedron and pentahedron elements as
defined by ESORT in *CONTROL_SHELL and *CONTROL_SOLID is now
considered when contouring element formulation

- **Case 20156**

Previous versions of PRIMER could contour element formulation but would
restrict values to a maximum of 100. This restriction has been removed. Any
element formulation value can now be contoured.

- **Case 17458**

If ESORT flag on *CONTROL_SHELL is set to 1 and the element form is set to
2 on the section card of a part made of shells, a plot of element formulation for
this part now shows 2 for the 4-noded element and 4 for the 3-noded elements.

Control

- **Case 21062**

In *CONTROL_SHELL, options 3,4 and 5 are available now for field LAMSHT.

- **Case 20937**

Removed option 4 and 5 for field SOLN on *CONTROL_SOLUTION, as manual
now has only options 0, 1 and 2.

- **Case 20322**

If LCTM is specified on *CONTROL_TIMESTEP, a model with large element
timesteps (or rigid elements only) will run at a value taken from the load curve.
PRIMER has been corrected to use this value.

- **Case 20111**

The drop-down for SWLOCL on *CONTROL_SOLID had the incorrect setting for local and global. Now fixed.

- **Case 19616**

Control_Implicit_Forming was not activating correctly on the control panel.

Curves

- **Case 21741**

Setting LCINT to a large number (e.g.10000) could slow down model checking considerably due to un-necessary repetition of the laborious discretization process. Now fixed.

Cut section

- **Case 20931**

When dragging a cut-section in translation the motion of the section follows that of the mouse, giving an intuitive result. However when dragging in rotation the direction of rotation could appear to go the 'wrong way' about 50% of the time. This was because of the overly simplistic approach used to turn mouse motion into plane rotation, and it has been improved so that dragging a section in rotation is now much more intuitive.
The way it works now is the initial mouse point is projected onto the cut plane, and all subsequent rotation is equivalent to dragging that point as a virtual 'drag handle'.

- **Case 20415**

In the cut section properties calculation there was an error in the calculation of the angle of the principal inertias (Iuu and Ivv) when the Ixx and Iyy values were identical but the Iuu and Ivv values were not. The values were correct, but the angle was wrongly reported as zero instead of 45 degrees. (Zero would be correct for a doubly-symmetric section where Ixx == Iyy == Iuu == Ivv, but not for the singly-symmetric case where Ixx == Iyy but Iuu != Ivv.)
This was caused by ill-conditioning in the Mohr's circle calculation for principal axes angle, since for Ixx == Iyy the radius of the circle is zero and special logic is required to obtain the angles of the principal axes.
This would not be a problem in normal usage since exact symmetry (to 7 sig figs) would not occur in 'real' sections, but it showed up in a simple test case of an equal angle section made from just two elements.
This has been corrected, and the 'special case' calculation now distinguishes correctly between the doubly- and singly-symmetrical cases.

Database

- **Case 20844**

On the *DATABASE_CROSS_SECTION edit panel, the '=' and '-' offset buttons would offset from the original clicked node, rather than the currect CT values when 'auto-create' was on. This has been corrected.

- **Case 20705**

You could not use the '+' and '-' offset buttons in the database cross section edit panel unless you had clicked on a node to define the starting position (i.e. you could not offset a typed in starting point, or a starting point defined by 'copy existing'. This has been corrected.

- **Case 21609**

Multiple *DATABASE_BINARY editing panels open at the same time were not working correctly. Now fixed.

Deleting

- **Case 21452**

The deletion of a part or part set is blocked if the part/part set was referenced as a connection layer. However, if the layer was defined by a name (part, part set, cad name or assembly name) deletion was not blocked. This has been corrected.

- **Case 21483**

PRIMER could crash if using 'cleanup unused' in a model containing INTEGRATION_BEAM definitions. Now fixed

- **Case 21225**

Previous versions of PRIMER would sometimes throw a spurious error message complaining about a missing test function for *RAIL_TRACK. This would happen during cleanup. This issue has now been fixed.

- **Case 19666**

'Forced' deletion should delete anything regardless of what refers to it, leaving a latent entry in its place if it is referenced by something else. However this was not working if an element was used in a connection, and 'No action' was chosen for connections.
This has now been fixed. Membership of a connection will not lock items in the forced case leaving these items latent. This results in the connection itself becoming 'bad', but this shouldn't matter in typical usage of forced deletion where items are replaced, as the new definition will make the connection 'good' again.

Dummies

- **Case 20619**

When using command-line input to drive dummy or mechanism motion the input syntax is:

/DUMMY Mi/j where i is the model number, and j the dummy number

However if there is only a single model with a single dummy (or mechanism) in the database the 'Mi/j' syntax is redundant because PRIMER automatically selects this dummy.
If 'Mi/j' is present in this situation a warning from the input command parser

appears, however the process continues to work correctly. So the warning does no harm, but it is alarming for users.

It appears because the input token 'Mi/j' is not necessary, so the command processing which would normally handle it is skipped and that input token is passed on to the next level of command processing which does not understand it. This has been fixed by flushing the command buffer when only a single model + dummy / mechanism exists in the database, meaning that this warning is no longer produced.

- **Case 20260**

When an occupant is moved bodily a check is made for any (positioning) 'points' declared as the H-Point to see if they still match the actual H-Point, and if they do not the point definition is updated to correct this.

When such points were defined by coordinates this worked correctly, but when they were defined by a node the test could fail in some circumstances, resulting in a perfectly good H-Point 'point' definition being over-written with information which was numerically correct, but which would lose the definition by node. The number (label) of the point would also tend to be changed.

This has been corrected. If an H-Point 'point' is defined by node the test considers the nodal coordinate, and only updates the definition is this no longer matches the declared dummy H-Point coordinate.

- **Case 19799**

Added a minimum part mass used during the automatic selection for LS-DYNA dummy positioning. With this on (which it is by default) PRIMER will ignore nodes an rigid parts below a specified mass (0.01kg by default). This will help when positioning dummies which contain small rigid target markers.

- **Case 19730**

If a database contained multiple models, only some of which contained a mechanism or dummy, and the user attempted to use the command line /DUMMY or /MECHANISM commands to select a non-existent model/dummy combination a crash occured. This has now been fixed.

Edit

- **Case 20967**

Text editing unlabelled items could fail as PRIMER would not find the correct keyword to overwrite when re-importing the changes. Special comment lines are now written when editing which now allow PRIMER to do this.

- **Case 20575**

Field N3 for elform 3 beams was being populated when using pick closest nodes option. Now fixed.

- **Case 19864**

PRIMER allowed *DEFINE_TRANSFORMATION cards to be created with the POINT definitions after the POS6P or ROTATE definitions. Checks have been added to prevent this.

FMH

- **Case 21233**

PRIMER now uses the names of standard positions/target points that are provided as input in the first column in a csv file for the FMH function (Tools->FMH). If non-standard position are encountered, they are marked 'USER' as was done in previous version.

- **Case 20505**

The interior head impact script has been modified to improve markup over overlapping entities.

Geometry

- **Case 18689**

IGES surfaces which used spline curves for trimming may not have displayed correctly. Now fixed.

Graphics

- **Case 21457**

If 'data' component 'Shell outward normals' is used, in which shells are displayed using magenta when facing the user and blue when facing away, the plot can go wrong if more than one model is present in the database. The second model may not be lit, some of its elements may be drawn very distorted and - in rare cases - there may be a crash.
Now fixed.

- **Case 21390**

Toggling the visibility of models on / off in Model List did not, in itself, cause them them to appear or disappear from the the display. Some other action, such as a drawing command, was required to do this.
In contrast using the up and down arrow keys as short-cuts to perform the same

action *did* cause the display to be updated.

This inconsistency is now fixed: using the buttons in Model List will now update the display immediately to show which models are visible.

- **Case 21292**

PRIMER has the ability to save graphics attributes information (colour, transparency, plotting mode) for parts in the keyword deck by adding PR_PART_COL comment lines. This means that when a deck is written out any specially set up colours etc are 'remembered' and restored when it is next read in. However it also means that a user who reads in a deck from an unknown source may also be importing non-standard graphics attributes, and this can be a source of confusion.

More significantly when a data-bearing plot (CT, SI, VEC) is performed then elements which don't have the default 'current' plotting attribute will get drawn in wireframe, hidden or shaded modes and - crucially - will not get contoured. This may be not only confusing but also potentially misleading.

To try to prevent this happening PRIMER now detects if a data-bearing plot is performed and one or more elements are not contoured because of their preset plotting mode, and issues a warning explaining what has happened and how to change it. To avoid annoying users this warning only appears once for a given model during a PRIMER session.

In addition the current method of changing or resetting plotting modes, via Quick Pick or via Display > Colour, are both flexible but require several mouse clicks. Therefore a new 'Reset attributes' button has been added to the Tidy popup menu which will restore all graphics attributes with a single click. By default the uper case 'K' keyboard shortcut will perform the same function.

- **Case 21180**

When creating or editing a rigid wall it is possible to drag the current definition using the cursor, and this gave rise to some confusing graphics.

An 'after image' of the wall in its pre-dragged position would persist on the plot until the view was changed or an explicit view update (eg LI, HI, SH, etc) was performed.

This has now been fixed, and dragging a wall will always show only its current, edited position.

- **Case 20727**

On a not too minimal build of CentOS 6.x all our graphical software crashed during initialisation.

This turned out to be because only a very minimal number of fonts were part of the basic CentOS 6.x build, and moreover at an unusual range of sizes in points. As a consequence the initialisation of fonts for graphics failed to find even a basic font, and the codes (which share common graphics drivers) all crashed.

The handling of the 'missing fonts' problem has been improved. Rather than looking for a limited alternative range of fonts it has a much more flexible search process that is far more likely to find suitable alternatives of approximately the right size. If it does still fail then at least it will now do so gracefully, giving an error message that explains what happened and also what to do about it.

Of course the proper solution is to load the appropriate font packages onto the machine.

- **Case 20605**

When a model contained *DEFINE_COORDINATE_VECTOR the graphics could go wrong in various ways if this was displayed.
This was due to setting the origin of the system to (0,0,0) when computing the 'bounding box' around the items to be displayed, resulting in the origin being a potentially random number.
Now fixed.

- **Case 20349**

When beams are rendered using 'true' sections it is possible that it will become impossible to screen-pick them if they have been drawn in shaded or contoured modes. This has been fixed.

- **Case 19849**

When a mouse has a scroll wheel this can be used to zoom the image in the graphics window in and out, with the centre of zoom being the mouse position. However on Windows platforms it was possible to zoom the graphics image even when the mouse was not in the PRIMER window, but elsewhere on the desk-top. Since zooming takes place about the mouse position this could result in strange and unwanted changes to the image.
A check for 'is mouse position inside the window?' has now been added for mouse scroll wheel events, and they are ignored if they fail this test, which fixes the problem.

- **Case 18446**

When rigidwall visibility was turned on the behaviour of (graphics) 'Autoscale' was somewhat crude. It took the largest of any wall side, length, diameter or radius dimension (regardless of wall type) and treated this as the radius of a sphere around the wall origin. Infinite walls did not affect scaling at all.
This overestimated the size of finite wall, resulting in the scale used being too small, and often also misplaced the model centre.
The scaling of infinite walls is unchanged, since there is no point in trying to draw an infinite object 'correctly'; however autoscaling of finite walls now calculates the wall dimensions correctly, giving a more sensible centre and scale.

Groups

- **Case 18845**

I/O of very large models with very large numbers of groups was found to be glacially slow. The logic has been improved to achieve a considerable speed up.

Include

- **Case 21715**

When a model contained a *SET_..._COLLECT definition then using the INCLUDE option to add a new child include file which also contained a *SET_..._COLLECT definition of the same label generated a label clash.

This was wrong: *SET_..._COLLECT is a special case that can occur any number

of times with the same label, either in the same file or in include files. The label clash checking used in this context did not consider that. Now fixed.

- **Case 20659**

When an include_transform is scanned, PRIMER was failing to read the include label range information. Now fixed.

- **Case 19395**

When PRIMER reads in a keyword file it normally performs a considerable number of 'post-read' checks in which things like unresolved parameters, bad topology and a host of other potential problems are sorted out.
However when an include file was added to an existing model these checks were omitted, resulting in inconsistent behaviour. It could also lead to errors going uncorrected.
This has now been fixed: reading an include file into an existing model now goes through the standard post-read checks.

- **Case 18271**

Keyout of scanned model could generate empty files. The function has been corrected so only the files scanned in will be keyed out (or a subset of them if select files mode is used). Further the master file will refer only to the files that have been scanned, unless the default option 'suppress reference to unscanned includes' is switched off. In this case the original unscanned includes will also be referenced.

- **Case 21593**

If the preference primer*include_file_method was set this could get reset to 'In sub-directory' (the default value) if a model was written that did not have any include files.

Include transform

- **Case 20113**

When writing a keyword file PRIMER now checks to make sure that no negative labels will be created when 'unapplying' *INCLUDE_TRANSFORMs.

- **Case 20112**

PRIMER could give errors about label clashes if *INITIAL_AXIAL_FORCE_BEAM cards were used in transformed include files. Now fixed.

- **Case 19753**

SET_..._GENERATE items did not get offset by include transform. They are now offset.

- **Case 19703**

Several include transform issues have been fixed.

- **Case 19173**

Orientation of *DEFINE_CONTACT_VOLUME keywords was not working correctly. Now fixed.

- **Case 21590**

When using 'Include file, replace contents' from the include tree an infinite loop could occur if:

The include file was an include _TRANSFORM and the new include file contained comments.

The symptoms would be a pause in which the PRIMER process would consume all the available memory on the system, and then freeze or crash.

Initial

- **Case 21579**

*INITIAL_VELOCITY_GENERATION can accept zero in the ID field, meaning all nodes in the model.

The keyword editor for this card would accept this, but the scalar editing panel would not.

Keyword

- **Case 21716**

PRIMER will now read the keyword *DEFINE_TRANSFORM as an alternative to *DEFINE_TRANSFORMATION.

- **Case 20990**

PRIMER will now read/write models containing *INITIAL_STRESS_SHELL cards with = 3, which is viable for type 17 solids with *CONTROL_SHELL = 2.

- **Case 20950**

The label for *DEFINE_GROUND_MOTION would not be stored correctly in PRIMER. This has been corrected.

- **Case 20784**

MAT_054 wrong field description - Corrected to SLIMIT2.

- **Case 20686**

'Update DEF_2_RG' will now be always greyed out if there are blank rows.

- **Case 20555**

When emptying the contents of a part through the part edit panel, you could get incorrect messages about elements being added to the part. This has been corrected.

- **Case 20498**

Previous versions of PRIMER would not permit multiple instances of the *CONTROL_FORMING_ONESTEP card in a model. This has been fixed.

- **Case 20395**

PRIMER could not read *LOAD_THERMAL_TOPAZ. This has been corrected.

- **Case 20303**

IPNINT can now have values greater than 1. PRIMER no longer flags that as an error. Also, the text popup associated with this field in the edit panel has been removed

- **Case 20165**

The field ETFLAG in *CONTROL_PORE_FLUID can now take the value 2. PRIMER will no longer flag this as an error. The text popup associated with this field in the edit panel has been removed.

- **Case 20164**

Wrong warning message about new field on MAT_173,while keying out.

- **Case 20155**

PID field in *INITIAL_PWP_NODAL_DATA is now treated as an integer to prevent inadvertent cleanup. This field is currently not processed by LS-DYNA.

- **Case 20106**

An error which prevented card 4 of *MAT_HILL_FOAM (MAT_177) from being read into PRIMER has been fixed.

- **Case 20001**

Field headers for *LOAD_THERMAL_VARIABLE_BEAM were wrong when writing a keyword file. Now fixed.

- **Case 19606**

Part cards can now be created with a TMID but not an MID.

- **Case 19554**

It was not possible to create *INITIAL_VELOCITY_GENERATION_START_TIME cards with stime < 1.0. Now fixed.

- **Case 19483**

Wrong error check for field on MAT_219. Now fixed

- **Case 19419**

The optional _DISPLAY card for *RIGIDWALL_GEOMETRIC was always expected by PRIMER when _DISPLAY was present. This has been corrected.

- **Case 19412**

When reading material data, and excess lines of data are provided in the input deck, then PRIMER will (correctly) terminate the input with the MISSING_DATA error.

However the line number reported could be wrong if the _TITLE suffix on the *MAT keyword was used, quoting an earlier line number, and this could make it difficult to determine the exact nature and location of the error.

This has been corrected: the line number should now be correct, and the error message has been amended to make it clear that missing data can also arise from excess data on a previous keyword fooling PRIMER into thinking that it is reading another definition.

Keyword editor

- **Case 21375**

The keyword editor suffers from the limitation that (for speed) it works on a 'template' definition of the items being displayed, and this template is fixed. Therefore cards with variable numbers of input lines tend only to show their fixed lines.

Two particular cases have been singled out for criticism since they are commonly encountered:

(1) Mat 34 (fabric) which may have extra lines depending on the value of field FORM.
(2) Section Shell and thick shell, which will have beta angles defined for field ICOMP = 1.

To try to solve these problems the material and section keyword editors now perform an extra pass once they have built the basic template for their definitions, which is to scan through all existing definitions of the relevant type in the model checking for the presence of extra data fields. If any such lines and / or fields are found these are added to the template, and thus become visible.

This is not a perfect solution since it means that if the keyword editor is used to create or modify one of the definitions above for the first time those extra fields will not be 'known about' at the time of launch, and will not appear. However they will become available if the editor is closed and re-opened since they will now exist in the model.

In most cases the model will already contain these definitions, so they will appear. However where they do not there is always the option of using the [right click] edit capability on the keyword editor row to launch the normal editor for that definition, and this will show all fields.

- **Case 21302**

Solid keyword editor broken - radio button (no. of nodes) doesn't work. Now fixed.

- **Case 20842**

The initial stress shell keyword editor could crash if any of the field names used more than 9 characters. e.g. if the number of history variables was > 9. Now fixed.

- **Case 19765**

Editor allows negative values for SREF and TREF on *INTEGRATION_BEAM.

- **Case 19681**

The keyword editor for *SECTION_BEAM incorrectly limited the input to the NSM field. Now fixed.

- **Case 19659**

NINT in *INITIAL_FIELD_SOLID can now also take values 2, 4, 5, and 14. A couple of issues with the keyword editor for this keyword have also been fixed. In particular, a spurious error message has been removed and the format fixed.

- **Case 19646**

A crash could occur in the Material keyword editor if:

1) The material type of one or more materials was changed
and
2) The material being changed had previously referenced one or more loadcurves.

Now fixed.

- **Case 18777**

The material keyword editor (Tools, Material, Keyword) has been criticised because it does not always shown optional data fields for materials in which the number and type of row / column data 'slots' that are defined depend on the content of the definition itself. A particular example is the case of MAT_FABRIC (34) in which setting FORM can cause extra data rows to appear.
This is an unfortunate consequence of how the keyword editor works. In order to give acceptable speed it sets up a 'template' of its contents, based on an empty definition, which dictates what row / column data slots will be shown and what they will contain. This is fixed, so the case of 'item grows if its contents change' cannot be accommodated because it would break the fixed template.
However a partial fix has been introduced. When setting up the material keyword editor it creates empty material definitions as before, then it also scans all installed materials of that type in the model to see if they contain extra data fields. If they do these extra fields are added to the template. In this way existing data in the model will be shown, which will satisfy most users.

This is not a complete fix since if the user changes a material definition inside the keyword editor in a way that should cause extra data fields to appear, but no existing material uses these fields, the fixed template will not 'know' about them so they won't appear. However it is at least an improvement.

- **Case 17735**

Keyword editing panel for *FREQUENCY_DOMAIN_ACOUSTIC_BEM is made available. At this point of time, it can take care of fixed rows only.

Labels

- **Case 21394**

Material->copy did not work if all of the materials in the model used character labels. Now fixed.

- **Case 20512**

PRIMER provides both 'static' labels, as turned on in the Entity panel for a category, and 'transient' labels as used by Quick pick and related items.
When more than one model is present the transient labels have (quite correctly) prefaced the item label with a 'Mn/...' model number prefix so that the user can tell in which model the item resides.
However if static labels are also turned on for that item then, by default, they will not add a model Mn/... prefix, meaning that when the transient label is superimposed on the static one both become unreadable.
To fix this this transient labelling now uses a more sophisticated algorithm to determine whether or not to add a model Mn/... prefix so that it will always match any static label for the item that is already present. This means that the two labels are now the same, so that when one is drawn on top of the other there is no visual change and both are still readable.

- **Case 19478**

PRIMER could occasionally give errors when using latent (referred to but not existing) character labels in editing panels. Now fixed.

- **Case 19424**

Under Model, Renumber includes, it is possible that negative labels will appear in the current label ranges for include files. This would occur if:

1) Include files contained items using character labels (valid for *SECTION, *MAT, etc)
2) Include files contained *SET_..._COLLECT definitions.

The reason is that in both these cases PRIMER assigns 'private internal' labels that are negative, and these should not be visible to users. However these negative labels were not being trapped and eliminated from the calculation of current label ranges in include files.
This has been corrected, however there remains the logical problem of 'what is the valid label range for an include file if it contains items with both conventional numeric and also character labels?'
There is no obvious answer to this question, so currently PRIMER ignores character labels when calculating these label ranges.

Loading

- **Case 20833**

Drawing of LOAD_RIGID_BODY has been corrected for cases where body is a nodal rigid body.

Local axes

- **Case 21069**

Previous versions of PRIMER would report incorrect values of local material angles in certain situations. This has been fixed.

- **Case 20594**

The local element display menu (Entities) previously included option buttons for LS-DYNA versions '971-R3' and '971-7600'. These would determine precedence between master part and part referenced in the integration rule for a particular integration point. This option was found to be confusing and has now been removed.

Macro

- **Case 21022**

Macro playback could crash if a radio button macro command was being played and the radio button could not be found for some reason. Now fixed.

- **Case 20284**

PRIMER could occasionally give an error 'ERROR in SM_TEXT_BOX_INPUT: Parent box not mapped' when a Pause command was added to a macro. Now fixed.

- **Case 17945**

When recording a macro slider movement would not get recorded correctly for large lists. Now fixed.

Mass

- **Case 21849**

Assign mass function has been enhanced to restore *ELEMENT_MASS_NODE_SET definitions on completion of a massing up operation. The process worked correctly for assign mass create/modify.

However, temporary element masses could be created and left behind at out of range labels when using assign mass recalculate, assign mass off part-replace or 'simplify and delete' rigidify. This has been corrected.

- **Case 21726**

Two bugs have been fixed which may affect calculation of nodal mass on rigid nodes which mesh to multiple parts. The lumped mass was not being partitioned correctly. Also the mass was not being zeroed if one of the parts is PART_INERTIA. Both could result in a small overestimate of the mass.

- **Case 20587**

The beam mass calculation routine has been enhanced to calculate the mass of integrated beams.

- **Case 19863**

A model built from database/template method in simple mode failed to set the 'data changed' flag on the new model. This resulted in a failure to calculate *ELEMENT_MASS_PART until an operation (such as update of an edit panel) trips the flag.

- **Case 19804**

Model with *ELEMENT_SHELL_COMPOSITE could crash due to an error in the function which calculates the thickness of the composite shell. Now fixed.

- **Case 19706**

Added mass on resultant beams which use the defined section formulation (d1,d2,etc) was being incorrectly reported as zero. Now fixed.

- **Case 21563**

ELEMENT_MASS_PART was programmed originally to be applicable only for shell parts where area weighting is applied. The keyword may now be applied to solid parts where volume weighting is applied. Primer has been updated accordingly

Materials

- **Case 21622**

Renamed a field in card 3 at column 4 to 'FLDCID' on *MAT_FLD_3-PARAMETER_BARLAT (MAT_190).

- **Case 21431**

Optional card 3 on MAT_224 now works correctly.

- **Case 20620**

When PGP encryption is used to hide material data PRIMER substitutes a material property full of zeros and saves the true encrypted properties for subsequent output.
This process was failing for material type 196 because of a legitimate but rather over-sensitive check when reading the variable number of rows of data that this material type can use.
The check has been scaled back to a warning, so that in the conventional (unencrypted) case the user will be warned if the wrong number of variable data rows is found, and in the encrypted case it will be read silently.

- **Case 20609**

An incorrect error check for aopt = 2 when vector D is undefined has been fixed.

- **Case 20091**

MAT_3-PARAMATER_BARLAT field was silently being zeroed while writing keyword file versions < 971 R5.
Also fixed, error while keying out field <M> for versions < 971 R5.

- **Case 20072**

MAT_41-50 can now be used with discrete beams.

- **Case 19796**

Card 3 now gets properly updated in *MAT_191, depending upon value of
FTYPE.

- **Case 19684**

Added units (stress) for fields in *MAT_HYPERELASTIC_RUBBER.

- **Case 18388**

MAT_172: -ve number for AOPT field is not supported.

- **Case 17168**

*MAT_133: Three cards are added for the case HARD=3.

- **Case 16197**

Historically MAT_USER (type 41) definitions which used loadcurves might
suffer because PRIMER did not 'know' that these loadcurves were referenced, and
a Cleanup Unused would delete them.
The behaviour now is that all MAT_USER definitions will reference all
loadcurves in the include file in which they are defined, thus locking them against
deletion. This is a 'hidden' reference that is internal to PRIMER, and which does
not affect the external representation of the keyword in any way.
This behaviour can be controlled by the preference primer*mat_user_ref_curves
which can have the values:

IFILE_ONLY (default) reference all curves in its include file.
ALL reference all curves in the model
NONE do not reference any curves.

Mechanism

- **Case 21288**

PRIMER could crash if you had incorrect mechanism data (maybe through a
manual edit of the keyword deck) and deletion was used in PRIMER. This has
been corrected.

- **Case 21113**

When creating mechanisms it is illegal to have 'unconnected' assemblies that are
not connected to anything else in the mechanism. Implicitly this also means that a
mechanism containing a single assembly is illegal.
When creating or modifying mechanisms this error would be detected and
reported, but the error message was a bit obscure and did not identify the
offending assembly. This has been corrected and each such illegal assembly is not
reported by label and name.
Attempting to position such a mechanism would also fail with another error
message (which has also been clarified), and a 2nd attempt to position the

mechanism could lead to a crash.
This crash arose from the original failure and has been corrected.

- **Case 19708**

Mechanism attributes can be saved to a Mechanism Configuration File (.mcf) and restored at any time during positioning. However this file relies on the names of connections and points when matching up the mechanism on file to the current one, therefore it can fail in two situations:

(1) If the names do not match then data will not be restored. This is unavoidable, and is not an error.
(2) If the current mechanism has multiple connections sharing the same name then data from disk would always get restored to the first matching connection in memory, resulting in the second not being updated. The same would be true for points.

This second point is more subtle, and the obvious workaround is to make sure that connections have unique names.
However it caught out one user, and therefore the reading of MCF files has been modified so that once a connection or point of a given name has been updated it will be ignored for any subsequent updates during that MCF read.
This solves the 'duplicate names' problem, but it would be better to avoid the problem in the first place by making names unique.

- **Case 19388**

Positioning a mechanism would always replace any user-defined title with the default 'Mechanism n'. This is an error.
The logic is now amended so that the title only gets auto-created in this way if it is currently undefined.

- **Case 16159**

It has been a criticism of PRIMER that when a PART is used in a mechanism assembly this locks the part against deletion, and that in order to delete the part one has to edit the mechanism first.
This is not an error, but clearly this behaviour annoys people. Therefore it has been changed and, as an exception to normal deletion rules, parts are no longer 'locked against deletion' if they are used in a mechanism assembly.
They will be deleted silently from any such assemblies, and it is possible that the assembly will end up empty as a consequence. This is not an error, although it would be unusual modelling practise, so a test has been added to the mechanism checking routine to detect this and warn the user if it occurs.
Note that membership of a DUMMY (as opposed to MECHANISM) assembly will still lock a part against deletion, and to delete such a part it will be necessary to edit the dummy assembly as before.

Menus

- **Case 21654**

On X11 platforms (i.e. Linux) the vertical axis label on a Curve graph in the curve or table editor could sometimes be drawn as a totally white rectangle, obscuring the text.

This was because non-horizontal text strings need special actions to render them, and these were using the wrong background colour in a few isolated cases. Now fixed.

- **Case 21426**

When an object menu in PRIMER is auto-expanded it becomes wider, but the buttons in the top part of it (Filter, Vis, etc) remain in the same place so that a user moving a mouse into the menu won't find that the target he was aiming for has moved.
This 'keep in the same place when expanded' logic is also applied to the Model (Mn) tabs used when the menu shows items from multiple models.
However model tab placement went wrong when in the original (contracted) state more than one row of tabs of was required, but in the expanded state enough width was available to display tabs in fewer rows. The Mn tabs remained in their original layout and obscured the lower row of Filter, Vis, etc buttons.
This has now been corrected. If multiple rows of Mn tabs are present in the contracted state, but fewer rows are required in the expanded state, then Mn tabs are re-arranged into fewer rows. This conflicts with the 'keep buttons in the same place' logic, but experiment suggests that using less vertical space - and thus making more space to display menu rows - is preferable to attempting to keep the original layout.

- **Case 21023**

Small menu windows were sometimes mapped 'too small', with scroll-bars where these were not necessary. This has been fixed.

- **Case 20938**

Object menus in 'Persistent' panels, typically those under 'tools', are given a size that tries to fit them into the available space without obscuring any buttons. In some cases the size and position of these menus was not being recomputed when the master window size was changed, resulting either in their obscuring things or being much smaller than they could have been.
Now fixed: object menu optimum size is now recomputed whenever mapped, taking into account the available space at the current master window size.

- **Case 18983**

If something which generated a 'stop and ask user' window that locks input occurred when the master PRIMER window was iconised it could appear that the user was locked out when he restored the master window.
This was because the 'locking' window could appear very small in the top left corner, and it would not be at all obvious that it needed dragging out to a sensible size in order to be clicked in.
Therefore if an iconise / restore sequence occurs on the master PRIMER window, and a sub-window which locks input is present during the restore process,

PRIMER will now always both bring it to the front and also make sure that it is a sensible size.

- **Case 18678**

The setting used on the element create panel were 'remembered' when the panel was dismissed and re-used when the panel next mapped when one might expect the defaults would be applied. Now the behaviour is configured by the user when the first element edit panel is dismissed.

- **Case 12018**

Selecting a new part for a layer in the connection table was very difficult as the part menu was too small. This has now been fixed by the menu auto expanding as required.

Merge

- **Case 21025**

With very large merge tolerances a node could be considered as a master node when merging one pair/group of nodes and a slave node when merging another pair/group of nodes. The node merge worked but undo would not work correctly. This is now fixed.

- **Case 21016**

Node merge could skip some nodes to merge if the merge tolerance was larger than the surrounding elements. Now fixed.

- **Case 20536**

Model merge would not do the options COPY_FROM_M1_OR_M2 and COPY_FROM_M2_OR_M1 correctly for types *DAMPING_GLOBAL, *LOAD_BODY and some of the *DATABASE cards. Now fixed.

- **Case 18287**

When doing model merge if both models had a title and the combined length of both titles was less than 80 characters then the combined title was used as the title for the merged model, otherwise 'M1 + M2' was used as the title.
Now if only one model has a title it is used instead of 'M1' or 'M2' in the merged title.

- **Case 17863**

Node merge is now significantly faster.

- **Case 17862**

The stop button can now be used to abort a very slow node merge (as can occur if the user typed in a very large tolerance)

Mesh

- **Case 21175**

When splitting multiple solids, the 'Split at ...' radio was not greying out, fixed this.

- **Case 20148**

Previous versions of PRIMER sometimes made it difficult to abort a multiple shell split operation if checks were turned on. This would happen if the new shells were in violation of any element quality check criteria. In that case 'Abort' would simply skip to the next selected shell.

'Create all' and 'Abort all' buttons have now been added as a workaround.

- **Case 20119**

PRIMER could, on occasions, crash while attempting to split shells. This has been fixed.

- **Case 19505**

New 'Only' button added to Meshing > Split > Warped to trias which blanks everything but warped shells.

- **Case 19067**

Fixed: single solid splitting gets wrong node labels

- **Case 18203**

When removing a hole it is now possible to use a planar surface when creating nodes. Additionally the algorithm has been improved when the shells around the hole are considerably out of plane.

- **Case 18165**

When splitting shells PRIMER used to calculate the ID every time whenever a new node or shell is to be created. This was making the process very slow. Now this calculates the starting node and shell ID once and after all the shells and nodes are created, it renumbers those based on the option selected.

- **Case 14416**

When splitting beams a unique 3rd node is generated for each split beam when NREFUP set

Model build

- **Case 20968**

Interior head impact build from csv requires that the headform local x axis (line of flight) aligns with +ve X. If this is not the case the build process is stopped with a warning message.

- **Case 20606**

With very large models, an annoying error message which reported that bucket sort setup was ill-conditioned could recur, even though the process worked adequately. The tolerance which determines when the message is given has been adjusted.

- **Case 19789**

Use of UNC path to specify 'rootdir' in model build from csv was causing build to abort even when the directory definition was correct. PRIMER11 has been fixed to allow this.

- **Case 19614**

When comparing the root directory and the filename directory on windows to determine relative path for keyout of an include file, PRIMER failed to allow for different capitalization of the drive. This resulted un-necessarily in a reversion to the full path.

- **Case 18438**

When component files contained parts with missing section and/or material cards, the 'fix latent' function of model build would delete the part itself to try to remove the problem. This is unlikely to be the correct solution, so the part is no longer removed but left with latent references which the user must resolve before submitting the job.

- **Case 18436**

Multiple model build on model containing set_generate definitions could be frustrated by repeated appearance of a panel requesting user input on whether or not to consider set upper bounds when determining the highest label for newly created items. The default for the preference that controls this has been changed from 'prompt' to 'ignore'.

Model modified

- **Case 21689**

"Model modified" did not find changes in part composite integration points. Now fixed.

- **Case 21179**

PRIMER gave an error when model modified was used in 'compare to original' mode with a model created in PRIMER. Now fixed.

- **Case 18726**

Model modified will now expand the include and red light any modified includes including those at a depth greater than 1.

Nodal Rigid Bodies

- **Case 20463**

'Undo last' on the NRB creation panel would also delete any referenced coordinate system, and anything that referenced that coordinate system. This has been corrected.

Node

- **Case 20615**

The node creation panel now allows picking of nodes from the other models, the coordinates of which can then be used to create nodes in the current model

- **Case 17846**

Node replace is now faster

Orient

- **Case 21737**

Orient into contact was not working with composite parts because these don't have a section card, but require thickness calculation from the layers. Now fixed.

- **Case 20876**

Orient->Project to mesh operation could occasionally fail while attempt to project entities along global X. This has been fixed.

- **Case 20508**

Copy orient on part used as Lagrangian container in *INITIAL_VOLUME_FRACTION_GEOMETRY incorrectly copied the initial card and the *ALE_MULTI-MATERIAL_GROUPS.

- **Case 20316**

The orient scale drag function has been improved to give a more 'natural' behaviour. The image should scale up now when the drag is away from centre (of scale) and scale down when the drag is towards it.

- **Case 20118**

The sequence of operations drag translate, accept, drag translate, undo did not apply the correct undo vector as the accumulated transform was not zeroed as it should have been after the accept. This does not affect nodal coordinates (as they are reset) but will affect coordinates in other contexts, e.g those of *DEFINE_VECTOR. This has been fixed. The (more common) operation drag translate, drag translate, undo was performed correctly.

- **Case 20071**

Orient into contact did not work well with flat plates. Some new logic has been added which should deal with this difficult case more successfully.

- **Case 20053**

The default propagation for orient of parts resulted in inconsistent treatment of associated items such as *BOUNDARY_PRESCRIBED_MOTION, *BOUNDARY_SPC, *LOAD. These were oriented only if they referenced a *DEFINE_VECTOR or *DEFINE_CSYS. Now by default they will not be oriented.

- **Case 19941**

Orient scale by drag has been fixed so that drag away from centre always scales up and drag toward centre scales down.

- **Case 19850**

PRIMER could get stuck in a loop when a model contained two assemblies which were children of each other. Checks have been added to prevent this from happening.

- **Case 19785**

The 'part only' option for orient would interpolate parts where any node has been selected. The logic has been changed to only interpolate parts which have an element all the nodes of which have been selected. Thus when elements on borders of parts meshed together are selected, the internal nodes of the unselected part will not interpolate.

- **Case 19047**

Orient->offset->sketch did not work. Three problems have been fixed.

a) Sketching while using offset option,
b) Corrected misleading button text.
c) Added an option to view the direction of offset.

- **Case 18893**

Action for reflection of DEFINE_SD_ORIENTATION vector used by rotational springs has been corrected. Additionally if reflection occurs and there is no vector or the vector is shared by translational springs, warnings will be given on orient.

Parameter

- **Case 21630**

PRIMER would think that parameters which were only referenced from *DEFINE_CURVE_FUNCTION strings could be removed when doing a 'clean up model'. Now fixed.

- **Case 21386**

The clipboard function 'Find referenced items' was not treating cases where *PARAMETER_EXPRESSION is referenced by a selected item correctly. Now fixed.

- **Case 20948**

When the cursor is hovered over a parameterised data field in a menu panel a popup is automatically mapped showing the parameter value and type, plus some editing options. If:

1) the parameter name is the full 9 characters and
2) the parameter value requires 9 or more digits and
3) a non-default larger text size is in use

The popup may be too narrow to display the full number of significant figures. The size of this popup is now adjusted to take into account these factors and it should always be wide enough to show all the data.

- **Case 20737**

When two models are defined such that:

Model A contains a reference to a parameter that is not defined.
Model B contains the definition of that parameter

A Model Merge of models A and B should generate a third model C in which the usage of the parameter is correctly resolved.
This worked if the order of the merge was such that the model containing the parameters came first, but not if it came second. In the example above merging models A and B would fail, but merging B and A would succeed.
This counter-intuitive result came about because inside PRIMER a 'merge' is in fact a 'copy' operation, and when a model is copied the very first thing to be copied over are its parameter definitions so that they are available for usage in the rest of the model. Therefore in the case of merging two models the sequence becomes:

(1) Copy parameters in model A
(2) Copy the rest of model A
(3) Copy parameters in model B
(4) Copy the rest of model B

Therefore it should now be clear that if a parameter is referenced in stage (2) above, but its definition is only supplied in stage (3), then it will not be resolved correctly.
This problem has to be confronted when reading a keyword input deck in which parameters may appear in any order, and PRIMER uses a post-read 'sort out parameters' process to go back and resolve missing definitions.
This sorting out process has now been applied following a model merge as well, with the result that the process is no longer sensitive to the order in which the models are merged.
This problem could crop up in other guises as well, notably the sequence:

Add parameters from model A into the clipboard
Merge into model B (where they are referenced)

This process also implied a merge, and suffered from the same error as the explicit merge. The fix corrects this situation too.

- **Case 20470**

You could not input a parameter to the BAMMG field of
*INITIAL_VOLUME_FRACTION_GEOMETRY. This has been corrected.

- **Case 20468**

Parameters were not being processed correctly during keyin of models containing
open-ended *CONTROL cards that used parameters. This has been fixed.

- **Case 20397**

Parameter association could be lost with fields in layer data lines for
*PART_COMPOSITE and *ELEMENT_SHELL_COMPOSITE. This has been
corrected.

- **Case 19405**

PRIMER V10 accepted PARAMETER _LOCAL and _DUPLICATION suffices,
but only did a half-hearted job of dealing with them. In particular it would not
tolerate duplicate parameter names even if the _DUPLICATION settings
permitted this.
PRIMER V11 now handles duplicate parameters in the same way as LS-DYNA,
permitting duplicate names to exist and using the instance dictated by the settings
on the PARAMETER_DUPLICATION card. Changing settings on this card will
result in the model being reconfigured to reflect the changed parameter usage.

- **Case 18102**

When originally introduced Parameters in LS-DYNA were theoretically limited to
7 characters in length, but testing revealed that they worked when 8 characters
long, so PRIMER did likewise.
The LS971 R6 manual now states explicitly that parameters can be up to 9
characters long, so PRIMER now accepts this from release 11 onwards.

- **Case 14034**

When reading some cards (in this example *AIRBAG_WANG_NEFSKE) in
which:

- Data fields use parameters
- Those parameters are defined later in the deck
- The parameterised data fields dictate how many lines of data are to be read

PRIMER would reject the input deck because data was missing. This was because
it would treat the excess lines as an incomplete 2nd definition of the same type.
It is now possible to read these decks in a limited range of circumstances as
follows:

(1) In the Model, Read Options panel select 'one definition per keyword' for the
item type in question. In this example that would be *AIRBAG, but the facility is
also available for other keywords liable to change, eg *MAT.
(2) Reread the deck.

This should now succeed, as any 'missing' lines are assumed to be be zeros, and
'excess' lines are ignored. Parameter values found later in the deck will be back-

substituted into the definition.
Note that the solution is not totally complete: if the newly found parameter values 'enable' the lines previously thought to be excess they will not be reread, but instead treated as if full of zeros. If there was non-zero data on these lines it will be lost.

Moral: define your parameters before they are used!!

Part table

- **Case 19776**

It was observed that the values on the part table sometimes change when refresh is pressed. A bug in logic which distributes lumped mass onto nodal rigid bodies which have off-structure nodes has been fixed.

- **Case 19467**

PRIMER could get stuck in a loop when the panel was small and the user was turning on a large number of columns. This has been corrected.

- **Case 14094**

Parts on the part table can now be sorted by model.

Part replace

- **Case 21318**

1) With MIG welds by default PRIMER will try to determine a new path along the free edge of the new part. This generally does work, sometimes there can be problems. There was a bug where it would not work if the lowest node label on the nodes around the free edge the MIG's are on was one of the nodes attached to a MIG weld. Now fixed.
2) With the 'find new MIG path' option off, the MIG welds just got emptied and not reprojected. Fixed so they are now reprojected.

- **Case 19589**

Part replace was failing to import 3rd nodes for beams being imported. Now fixed.

- **Case 18311**

A potential crash after part replace seen in models with set_general/generate definitions, has been fixed.

Part tree

- **Case 20511**

Parts with no elements were not displayed correctly in the part tree when in assembly mode. This has been corrected.

- **Case 19609**

An error caused by an uninitialised variable in the part tree has been fixed.

Penetrations

- **Case 18633**

For surface-surface contact gaps were not being reported between parts of the same type. The exclusion of same part interactions is now only applied to single surface contacts.

- **Case 16167**

By default the contact checker now runs in MPP mode. However the MPP LS-DYNA threshold for penetration report is hard-wired to 0.001 which will not suit all models. By default, PRIMER will use an automatic threshold of minimum of 0.001 and 0.005*characteristic element length. However the user may configure PRIMER to use 0.001 (or any other value) should he so wish. Input of zero will revert to the automatic threshold.

Quick pick

- **Case 21455**

PRIMER could change the main quick pick setting to 'Locate in tree' when using 'Locate in tree' from a right click. Now fixed.

Read

- **Case 21650**

On Windows platforms only the sequence:

- Use file filter to select filename to read keyword input file.
- Apply Read file.
- Use file filter again, but this time Cancel without selecting a file
- Apply Read file.

Would result in the file not being read in for a second time. This has been corrected.

- **Case 21342**

PRIMER could hang when reading a keyword file containing *NODE_MERGE. Now fixed.

- **Case 21114**

When using scan to read a model that contained parameters and include transforms PRIMER could duplicate model comments each time an include was read. If the model contained lots of include files this could cause PRIMER to run out of memory. Now fixed.

- **Case 20625**

PRIMER can now read *CONTROL_MPP_DECOMPOSITION_DISTRIBUTE_ALE/SPH_ELEMENT as an alternative to *CONTROL_MPP_DECOMPOSITION_DISTRIBUTE_ALE/SPH_ELEMENTS , and *SET_MULTI_MATERIAL_GROUP_LIST as an alternative to *SET_MULTI-MATERIAL_GROUP_LIST

- **Case 20500**

A new option has been added for reading keyword files that allows PRIMER to skip leading '0' characters from fields that are not correctly aligned and 'spill over' from the previous field.

- **Case 20410**

It the title part of an I10, A70 line contained commas, PRIMER could incorrectly read the label field. This has been corrected.

- **Case 19588**

A rare crash could occur if reading in a model containing parts with negative labels and a PRIMER part colour comment line. The only potential way of ending up with a part with a negative label is if the part was in an include transform where the label offset for the part is higher that the part label. Now fixed.

- **Case 18441**

Depending on the input settings in force PRIMER may read an input deck successfully despite errors. Typical examples might be duplicate keywords (label clash) in which the 2nd and subsequent definitions are ignored.
PRIMER reports these errors to the dialogue box, and they can be examined in detail either by scrolling that back or via the 'View log' option in the Model Read panel. However it has been criticised because the error messages may not always stand out sufficiently clearly, and therefore may be missed by a user in a hurry. Therefore if an input deck is read despite errors PRIMER will now put up a 'stop and warn' style panel with a message, forcing the user to acknowledge this before continuing.
Users who wish to demote this to a (new, and more obvious) warning in the dialogue box that does not require explicit acknowledgement can do so via a preference.

Renumber

- **Case 20430**

In heavy cases the renumber into include range function sometimes requires 2 passes to achieve a complete solution. The logic has been improved so that available labels are freed up before each include is renumbered.

- **Case 18178**

Using the detailed renumbering panel for a type (e.g. NODE) from the 'Renumber contents' panel was extremely slow for large models. It is now considerably faster.

Rigidify

- **Case 21551**

Rigidify function could convert discrete beams to rigid material which caused an LS-DYNA error. These are now flagged for deletion if selected amongst the parts to rigidify. If not deleted they will be left unchanged.

Scripting

- **Case 21842**

The "exists" property did not work for the History class. Now fixed.

- **Case 21623**

After doing Model.Merge() the merged model graphics (the clipping planes in this case) could be incorrect until an autoscale was done. Now fixed.

- **Case 21393**

Material.GetFromID() did not work for character labels. Now fixed.

- **Case 21304**

The Xrefs.GetTotal() method did not work if type was 'SBELT Defn'. This is now fixed.

- **Case 21280**

Model.Read(), Model.Write() and various other methods that use filenames now accept names in Japanese and other Unicode characters.

- **Case 21007**

System commands would not work if they contained Unicode characters. Now fixed.

- **Case 20898**

If a script was UTF-8 encoded empty strings could cause the script to terminate in some functions. Now fixed.

- **Case 20732**

The Add() member function of Set class in JavaScript may fail to work correctly. This has been fixed.

- **Case 20730**

File.FindLineStarting() could sometimes fail. Now fixed.

- **Case 20534**

The IntegrationBeam and IntegrationShell constructors could put junk data into the integration point data if NIP was not zero or junk data into the standard cross section data if ICST was not zero. Now fixed.

- **Case 20533**

The JavaScript constructor for *INTEGRATION_BEAM would not return the correct object. The entity would be created correctly, but if you had more than one of these cards in your model, the returned object from the constructor could be wrong. This has been corrected.

- **Case 20525**

When using model.user_data to save data into a keyword file lines longer than 80 characters would get corrupted. Now fixed.

- **Case 20292**

The method Set.AddCollectChild() did not work correctly. Now fixed.

- **Case 20259**

When importing an include file using Model.ImportInclude() PRIMER did not update the graphics correctly resulting in the new include file not showing free edges. Now fixed.

- **Case 19803**

New database and load body card properties were not accessible in JavaScript after importing/merging a model which contained database and/or load body cards. Now fixed.

- **Case 19802**

New control card properties were not accessible in JavaScript after importing/merging a model which contained control cards. Now fixed.

- **Case 19660**

Visibility('ELEMENT', false) did not work in 10.2. Now fixed.

- **Case 19533**

Right clicking on a combobox did not call the 'onClick' function (but left clicking did). Now fixed

- **Case 19484**

The SetEntity() function in JavaScript for finding attached items did not work correctly. Now fixed.

- **Case 19473**

Some menus in the pedestrian markup script have been made non-modal meaning the user can interact with the model and other menus whilst they are displayed.

- **Case 19453**

The part member function MaxMin() may not return the correct max/min values without a graphics update first if nodal coordinates had been updated. Now fixed.

- **Case 19244**

It was not possible to set optional fields on materials created in JavaScript (e.g. FOR on *MAT_68). Now fixed.

- **Case 17999**

Create() and Edit() methods have been added to the following classes: ExtraNodes, RigidBodies, Rigidwall, VelocityGeneration, Transformation

Seatsquash

- **Case 20675**

PRIMER would add all dummy parts to the INTERFACE_SPRINGBACK card created by DYNA seatsquash that is used to produce a dynain file. This has been modified to only add parts the user has chosen to remain deformable in the dummy.

Sections

- **Case 20036**

When ICOMP was set to 1 on section edit panel, PRIMER unconditionally changed a zero setting of NIP to 2. If the section is of membrane formulation the default is in fact 1. The section editor has been fixed to leave zero as valid setting for NIP. LS-DYNA will find its own default.

- **Case 18408**

The section editor allowed you to use STYPE, D1, D2 etc for *SECTION_BEAM elform 13 which was not valid. This has now been fixed. Additionally an error check has been added to prevent this error in future.

Selection

- **Case 21345**

At the request of a user type 'Database History' has been added to the items that constitute an 'entity' for selection and screen-picking.

(Generally speaking 'entity' is intended to include a range of physical - loosely 'structural' - items, but its use has extended over time to include other types, and this is just the latest of these.)

- **Case 21298**

When clicking 'None' in the selection menu, only the displayed (maybe filtered) list was cleared (this is correct), but the calling panel would then ignore any selected items not displayed in the current filtered list. This has been corrected.

- **Case 20952**

When using 'Keyin' in object menus it is sometimes necessary to add a prefix. For example if the input is of the generic type 'Element' then in order to select Shell 10 you would need to write S10.
This prefix is a standard PRIMER acronym of up to 4 characters, and its use in this context would fail if it contained anything other than the characters A-Z. Unfortunately this includes the case of sets, where a context using the generic type SET requires prefixes such as S_NO for a node set. This fails because these standard acronyms contain the non-alphabetic underscore '_' character.
This was due to an error in the logic used to parse prefixes, which assumed that they would only be A-Z. Now fixed.

- **Case 20951**

When using object menus to make a selection the sequence:

Drag out a screen area to select multiple items.
Then click on 'Scr Area' in the 'Visible' menu

Results in the 'Cancel + Help' panel occupying the whole screen. Now fixed.

- **Case 20574**

The logic has been modified for the selection menu on the Remove->Delete panel so that the current entity type you are deleting is retained in the selection menu after deletion rather than returning to the start point of a selection menu listing all entity types in the model. This is now consistent with the rest of PRIMER.

- **Case 20573**

PRIMER permits screen-picking to be carried out in 'Feature line' mode, in which a pick propagates out to all such items on an area bounded by feature lines defined by a specified angular change in the mesh normals.
This worked for selection using the left mouse button, but when the right mouse button was used to deselect it only removed the item under the cursor, and not all items bounded by the feature lines.
This has now been fixed: right mouse click in feature line mode is now deselects all items that would be selected by a left mouse click in that location.

- **Case 20502**

A selection list for *CONSTRAINED_LAGRANGE_IN_SOLID did not display any appropriate titles. This has been corrected.

Sets

- **Case 21388**

When editing a *SET_..._ADD definition the user would ADD sets of the same type to the 'parent' set. This could cause problems with duplicate labels if the default labels suggested by PRIMER were used because:

+ Top level set would be assigned a label <n> in the editor.
+ This set <n> would not yet exist in the database
+ When creating a child set PRIMER would offer label <n> as the next label.
+ Upon creation an error would be generated because the set would be self-referential.

To correct this the SET editor now makes an immediate latent (but unreferenced) entry in the database when a new label is defined for a set in the editor. This effectively occupies that label 'slot' meaning that PRIMER's normal 'give the next label' logic will not choose it.

If the edit proceeds normally to a successful conclusion then this latent entry will become permanently installed, otherwise it will be removed again.

- **Case 20955**

The cross reference viewer now shows the actual things referenced for GENERAL, ADD and INTERSECT sets as well as the 'contents' of the set (i.e. the items that finally end up in the set definition).

- **Case 20450**

PRIMER10.2 was incorrectly writing SET_LIST_GENERATE for all sets of type _GENERATE. The _LIST descriptor should in fact only be used for sets of type NODE and SHELL. Some versions of LS-DYNA may appear to read SET_BEAM_LIST_GENERATE but will in fact treat as a list set.

- **Case 18087**

PRIMER could not sketch a *SET_NODE_ADD set which referred to a *SET_NODE_ADVANCED set. Now fixed.

Shortcut

- **Case 20138**

The 'K' keyboard shortcut now resets entity label/name display settings to default as well as the entity display setting.

Sketch

- **Case 21201**

PRIMER could get sketching wrong, and give warning messages that should not be given with DATABASE_CROSS_SECTION cards using the field. This has been corrected.

- **Case 19587**

Sketching of *SET_SEGMENT_GENERAL did not work for cases where segments needed to me shown on shell elements. Now fixed.

Text

- **Case 19862**

If the preference setting primer*include_file_method was set to SELECT_FILES, the text editor failed to work. Now fixed.

Timestep

- **Case 21612**

The timestep calculation for orthotropic materials was not allowing Poisson ratio greater than 0.5. Now fixed.

- **Case 20538**

Timestep calculation for MAT66 discrete beams has been corrected to account for cases where abs(scoor) is greater than 1.

- **Case 21612**

The timestep calulation for orthotropic materials was not allowing Poisson ratio greater than 0.5.

Translators

- **Case 18294**

Beam element definitions are read in properly even if the part is not defined for those elements.

- **Case 21550**

/SURF/PART cards from a block format RADIOSS input file could be incorrectly translated if there were more than 3 parts. Some of the parts would be missed.

Undo

- **Case 21117**

Coat Undo was causing crash in PRIMER, now it is fixed.

User interface

- **Case 20899**

The xrefs, include, check and copy buttons were missing in some of the frequency editing panels. They have been added.

- **Case 18140**

The title button for 'TERMINATION' panel was missing. Now fixed.

Write

- **Case 21631**

When idle auto-timeout is used a dump of any models in memory is written to a keyword file in the specified directory. However the file name currently is of the form

saved_model_n.key (where n is the model number 1, 2, 3, etc.)

This means that successive timeouts will overwrite files from previous ones.

To solve this problem the date and time have been added to the filename so that it will now be, for example:

saved_model_1_Mon_Mar_04_15_25_37_2013.key

- **Case 18219**

During keyout, if drive mappings are set, PRIMER will now consider UNC paths if they have been used in the model read in.

2.2   D3PLOT

2.2.1   Bugs Fixed in 11

Ale or Euler

- **Case 17573**

2D Axisymmetric ALE analyses were not recognised as 'ALE', so the extra shell data components were simply treated as simple scalar values and not as ALE data.

This has now been fixed. If an analysis has ALE materials, but no solid elements, then it is treated as '2D ALE' (implicitly axisymmetric) and the special ALE components are plottable.

Annotation

- **Case 21230**

When the max/min in a plot is computed this could become 'out of date' if plotting modes are swapped between VELocity plots and contour plots of nodal data. The max/min for the velocity vectors would be displayed regardless of the nodal data being contoured.

Belts

- **Case 17728**

Two separate sets of problems arose with seatbelt-related element results in D3PLOT version 10:

(1) When 2D seatbelts were used the results for sliprings and retractors would come out as zero.

This was because 2D seatbelt output from LS-DYNA has mutated during the life of D3PLOT 10.0. These belts are managed inside LS-DYNA by creating parallel rows of 1D belt elements, sliprings and retractors; and the original output from earlier versions reflected this, giving '1d' results for these parallel rows.

Later versions of LS971R5 then changed to produce output for a single retractor or slipring at a given location, regardless of the number of elements across the belt. 1D belt element output was also deleted, being replaced with spring-like output to beams coincident with the 1D belt elements.

As a result D3PLOT 10 will not display 2D retractor and slipring results from later releases of LS971R5, since it is expecting data for parallel rows of 1D items.

This has been corrected, but the correction lies within the ZTF file created by PRIMER, and an up-to-date version of that is required in order to generate ZTF files formatted correctly.

(2) A separate problem, unrelated to whether belts were 1D or 2D, was that the export of max/min results to WRITE or XY Plotting was not working correctly for seatbelt-related items in the 'other' plotting mode.

This has been fixed.

Contacts

- **Case 20095 and 19524**

Two problems with contact force file (CTF) handling have been fixed:

(1) When a contact had no segments in the slave or master side then predictive picking would crash when it tried to draw a non-existing display list for the side with no segments.

(2) The internal format of the CTF file changed during the life of LS971 so that the internal node indices used to define segment topology no longer matched the same indices in the PTF file. Since D3PLOT assumed that the two were the same this resulted in the segment topology going very wrong. D3PLOT now detects this differences and resets the internal topology of contact segments so that their indices revert to being consistent.

Contour

- **Case 20141**

D3PLOT could make an error when contouring animations of shell parts in SI, CT or CL modes. This would only happen in the following circumstances:

- It is a shell part

- Animation contour scanning considers 1st and last states only

The error would be that for states between the 1st and last the contour bands could be wrong.

A workaround is to change contour scanning to consider all states, which will eliminate this problem.

- **Case 19858**

Turning on the 'Ref Node' switch in the 'Fix Node' panel did not update the contour plot. This has been fixed.

Turning it on in the 'Ref Node' panel worked.

- **Case 20595**

If a model contained only Thick Shell elements and these elements had been written out to the PTF with extra data variables at each integration point the option to select the 'Extra' data in D3PLOT wouldn't have been available. As long as the model contained at least 1 shell element then the option would have been available.

Additionally, a new option has been added to contour thick shells using either an 'interpolated' or 'simple' method. The 'interpolated' method is the same as previous versions and contours the shell from top to bottom, irrespective of the surface selected. The 'simple' method contours the shell based only on the surface selected. The method can be selected in Contours->Option->Tshell opts.

- **Case 20845**

The following sequence caused D3PLOT to not update a contour plot in a second window:

1. Load a model into window 1

2. Load another model into window 2

3. Window->Edit window->W1

4. Deselect M1, select M2 and Apply

5. Window->Edit window->W2

6. Deselect M2, select M1 and Apply

If you did a contour plot and then blanked items in window 2, the contour plot did not update.

Cut section

- **Case 20897**

D3PLOT would crashed when trying to display a cut-section if some coordinates in the results database were NaN (the IEEE special code for 'Not a Number').

D3PLOT now has a 'clamping' option which, when data are read in, will detect coordinates outside a user-defined bounding box around the model and clamp them to zero. This is intended to stop 'shooting nodes' from creating ridiculously large coordinates.

The same logic also detects NaN in the results and replaces it with zero, and this will solve this problem.

- **Case 15914**

When dragging a cut plane in translation the motion of the plane follows that of the mouse, giving an intuitive result. However when dragging a plane in rotation this was not the case, with the rotation direction being arbitrary. Rotation drag now gives intuitive results with the effect being as if the mouse had grabbed a 'Handle' on the plane and dragged it in the relevant direction.

- **Case 18796**

Cut-section true shell section thicknesses were being drawn as zero for *PART_COMPOSITE parts if the thickness data was not output in the PTF file and was being read from the ZTF file.

- **Case 20987**

In the cut section menu, if you entered a value of 0.0 for the thick cut thickness, D3PLOT would automatically turn the thick cut option off. However, it was impossible to then turn it back on again and enter a different thickness.

Data access

- **Case 20202**

Occasionally a plot of plastic strain or stress tensor derived data would show the wrong values at a given time, and after this D3PLOT could crash.

The problem could arise when stress or plastic strain data over multiple integration points was extracted. A typical cause of reading data for multiple integration points is plotting data for 'MAX_ALL' or similar shell surfaces.

The workaround prior to V11 is to make sure that data for all integration points of the given data component is loaded into memory in a single pass, and the easiest way to do this is to select surface MAX_ALL and to animate.

- **Cases 19039 and 19193**

A case was encountered where automatic conversion from IEEE 64 bit format data files to the native 32 bit internal format used for numbers inside D3PLOT produced some 'rogue' very large values.

While fixing this the opportunity has been taken to add a generalised 'clamp values to lie within a maximum magnitude' function to D3PLOT. This is optional, and is not used by default, but if it is selected then all data values (but not integers such as labels) read from disk have their magnitude tested against this value, and can be clamped to that value or reset to zero. This allows decks that contain a few spurious and very large values to have these reset so that they do not obscure data elsewhere.

- **Case 20149**

Some models could show very small values of PEMAG (plastic strain magnitude) when there was no effective plastic strain.

This has been fixed. If there is no effective plastic strain PEMAG will be zero also.

Database

- **Case 20055**

D3PLOT could crash when reading a reordered database with an embedded ZTF file. This happened in the rare case of the ZTF file having been written from a machine with a different architecture to the one writing the reordered database.

The ZTF file format and machine architecture are now compared and, if different, the ZTF data is not written to the database. A warning message is given to the user and instructions to write out a new ZTF file from the same machine.

Deform

- **Case 14809**

Magnify, Fix Node, Shift Deformed and Reference geometry in the Deform menu had no effect on airbag particles.

- **Case 19885**

Beams could appear to twist if true sections and magnified displacements were on and the model had been run with the *CONTROL_OUTPUT option NREFUP on (to update the coordinates of beam 3rd nodes).

Error trapping

- **Case 20123**

If D3PLOT crashed the 'Error recovery' panel used to offer to save 'a key data' but it was unclear what this data was. The 'key data' that is saved is a properties file for each model; this means that colour, blanking, transparency and display mode information are saved for (optional) reuse.

Femzip

- **Case 20972**

When reading multiple models compressed using FEMZIP the code worked correctly so long as each model was in its own window.

However all sorts of problems occurred if a FEMZIP-derived model was added to an existing window: coordinates in both new and existing windows could be corrupted, and the code would sometimes crash.

- **Case 21013**

When reading FEMZIP models there is a separate reader process for each model, and the main D3PLOT client treats these as 'server' processes that supply data on request. So have made a request it has to wait for it to be serviced.

This could take a significant amount of time for a large model, and if the user clicked on STOP (a 'user halt') this was treated as 'terminate the read process immediately'.

This worked, but D3PLOT saw this as a 'read error' and filled the relevant data vector with zeros. If this happened while reading nodal coordinates during the initial building of an animation it could result in the state at which the STOP occurred 'disappearing' because its coordinates were all (0,0,0).

Therefore the user-halt logic during FEMZIP read has been modified as follows:

+ A click on STOP during a 'wait for FEMZIP' is ignored until at least a 10 second delay has elapsed. Most reads take less than this length of time so the STOP will be handled elsewhere.

+ If it is then detected a message confirming that the STOP has been detected is issued, and the user is told that the user-halt will be deferred until the read process is complete.

+ He is also told that two further clicks on STOP will be treated as a read error and terminate the read immediately.

It would be a rare data read process that took more than 10 seconds, so hopefully users will never see this dialogue, and the user halt will be detected and processed safely in the main part of the programme once the data read operation is complete.

- **Case 15975**

Handling multiple Femzipped databases into D3PLOT was very slow because of the way the femzip decoding library works. To resolve this problem the Femzip reader has now been rewritten completely to run as a separate process for each model. In addition use of the latest 'extended' femzip library also gives faster access to data.

The combination of these two means that data extraction for a single Femzip model is now considerably faster than before, and the performance for multiple models is enormously (somewhere between 10x and 100x) faster.

General

- **Case 19456**

If a database file that doesn't contain any parts, e.g. a model containing only nodes, was opened D3PLOT could have crashed while trying to draw the Part Tree.

Graphics

- **Case 20453**

On ATI graphics cards only, plots containing triangular shells that were rendered with 'smooth' shading could show a 'splotchy' effect with dark blobs.

This was because a short-cut used to speed up rendering worked on other hardware, but not ATI.

- **Case 21308**

If hardware-accelerated shading (Tune, Use shaders) is not in use then D3PLOT has to generate outward normals for each 2d and 3d element facet when drawing shaded plots. These are expensive to calculate so they are cached for reuse, however operations which change the visible shape or geometry of elements require them to be recalculated.

One such an operation is [Deform] Magnify Coordinates which can change nodal positions when drawn. For small deformations slight errors in the outward normal vector are not noticeable, but

- **Case 19411**

If the background colour was white and the entity switch for drawing nodes was on then they would be correctly drawn in black. However, if a target marker was added to the model all the nodes would change colour from black to white.

- **Case 18097**

If smooth shading is selected (under the Display, Lighting menu) then reading a 2nd model into a contoured and lit window could result in the 2nd model's shading going wrong.

A similar problem could occur when swapping between flat and smooth shading when an animation was in progress.

- **Case 19407**

If (and only if) perspective was turned on then using the mouse scroll wheel in the graphics window to zoom at a point would result in the model 'walking off the page', or becoming invisible. The same problem would occur if the keyboard short-cut keys '+' and '-' were used to zoom in/out at the current cursor location.

Both functions would work correctly if orthographic (non-perspective) viewing was in use.

- **Case 19042**

The near and far (Z) clipping planes for the viewing frustrum are clamped to 'reasonable' values in order to prevent ill-conditioning during depth buffer calculations, and hence errors in the display of graphics.

However this clamping was too tight with the result that models with very large factors on displacement could end up being clipped by these planes. (The effect is like slicing off elements very near to and very far away from the viewer's eye, and is particularly noticeable when the image is rotated.)

These limits have been relaxed in order to make this less likely to happen.

Groups

- **Case 21430**

Deleting groups using GROUPS->DELETE could sometimes delete groups that hadn't been selected. This has been fixed in v11.

Image

- **Case 20934**

If an image was captured at 2 or 4 times the screen resolution the image was not drawn correctly if the option to change the window size in the 'Display Options'->'Window Dressing Menu' was changed from the default 'Full Screen' option to any of the other options.

- **Case 19887**

When 'tuned' graphics was active it could sometimes be the case that images captured to file (jpeg, png, pdf, etc.) would 'lose their lighting'.

Measure

- **Case 18694**

In the measure panel the results box was not always wide enough to show the whole number so it could display a value like '2.308E-' instead of '2.308E-4'.

In v11 the docked are can now be dragged wider by the user allowing the results box to be wider and hence able to fit the whole number in. Hover text has also been added.

Menus

- **Case 20140**

If the number of X-sections in a model was greater than the number of Parts then the blank menu wouldn't blank x-sections correctly when they were selected.

- **Case 20458**

On Windows 'File->Print' was starting the page setup menu and 'File->Page Setup' was trying to open the what's new PDF file.

- **Case 20685**

If the Advanced Layout option was used then when a new window was opened it could get created on a page behind the existing windows and wouldn't be seen by the user.

D3PLOT now checks the layout of the current page to see if there is any space on it when a new window is created.

If the current page layout is 'Tile Wide', 'Tile Tall' or 'Cascade' then the new window is always added to the current page.

If the layout is one of the other options then D3PLOT checks if all the slots on the page are full. If there are any gaps then D3PLOT adds the new window to the page. If there aren't any free slots then D3PLOT now checks the next page and so on until it finds an empty slot.

If the window isn't added to the current page because there aren't any slots then D3PLOT changes the active page to the one where the new window has been added so the user can see it.

Part tree

- **Case 18293**

D3PLOT could crash when updating a model in the part tree. This was only seen on Windows 64bit machines, but could potentially have occurred on other machines too.

Preferences

- **Case 20678**

Reloading graphics tuning preferences (D3PLOT*gtune_xxx) from the oa_pref file could cause a crash on start-up if the preferences had been captured on an old machine but were replayed on a new one, or vice versa.

- **Case 21478**

A model containing 297000 parts - a highly unusual situation - was extremely slow to open in D3PLOT.

Investigation revealed that the delay was caused when building the Part Tree, and occurred because the parts did not appear in ascending label order. The Part Tree displays in ascending numeric order by default, and the algorithm used to sort it into this order was very slow for large numbers of parts.

This has been fixed by using a different method of sorting part labels in that context.

Quick pick

- **Case 20929**

When using the 'Find' Quick Pick option to identify a solid or thick shell element (by keying in its label) this would fail to produce a result if the element in question was fully internal to the mesh and internal faces were not being displayed. This was because it was not considered to be visible as it had no exposed faces.

This has been corrected: normally 'find' will place cross-hairs on the first visible faces of a 3D element, but if there is none then it will now calculate the element centre and place the cross-hairs there.

Read

- **Case 19121**

D3PLOT will now read files correctly for models containing *ELEMENT_DISCRETE_SPHERE definitions.

- **Case 20671**

D3PLOT could not read it in a file if the TET10 flag on the *CONTROL_OUTPUT was turned on..

Scripting

- **Case 20975**

The GetData function wasn't working correctly when trying to get data for LSDA components. For example GetData(XSEC_M, XSEC, 1) should have returned the moment in the first database cross section, but it was actually returning the force.

The problem was that it was always returning data for the first component defined in the D3PLOT.components file. For XSECs this is force.

Similarly, for other LSDA entities (Spotwelds, springs...) the GetData function would always return data for the first component in the D3PLOT.components file for that type.

- **Case 20979**

The JavaScript function GetData() would return an error when trying to extract data for SOLID elements for the following components: S2MAX, S2MIN, E2MAX, E2MIN and ERATIO.

Sections

- **Case 18799**

The orientation of beams were not being drawn correctly if the orientation had been defined with the *ELEMENT_BEAM_ORIENTATION card rather than with a third node on the *ELEMENT_BEAM card.

This orientation vector data is now written to the ZTF file and used by D3PLOT to calculate the correct orientation of the beams.

T/his link

- **Case 18181**

A problem with contour colours not being updated correctly when non-default colours were used has been fixed. This could happen if windows (or models) had been deleted, resulting in existing windows being renumbered downwards.

- **Case 19374**

If when running the D3PLOT->T/HIS link the option to draw the locate symbols as circles was used then the circle size displayed didn't always match the size requested by the user.

- **Case 20702**

If a model with a full pathname longer than 200 characters was read into D3PLOT then T/HIS could crash if the D3PLOT->T/HIS link was used. Version 11.0 had been changed to support pathnames of up to 256 characters across the link.

- **Case 21460**

If an image was captured while running the D3PLOT > T/HIS link then D3PLOT could crash if the number of T/HIS graphs was more than the number of D3PLOT windows.

User interface

- **Case 21072**

The Utilities > Model title menu allows you to change the title displayed in a graphics window for a model, but it only permitted changes to the first model in a window. Where a window contained multiple models it was impossible to change the title shown for the 2nd and subsequent models.

This has been corrected: all model titles for the first active window in that panel are now shown, and any can be changed. If multiple windows are present and active in the panel these changes are applied to all active windows that contain that window / model 'slot'.

- **Case 19620**

The Apply button would stay greyed out if an existing file was selected in the Save Settings panel and then the 'Retrieve Settings' button was pressed before selecting either 'New Filename' or 'Delete'. This has been fixed.

User defined

- **Case 21747**

If a solid/shell user-defined data component was read from an ascii file then a crash could occur if:

- the requested shell surface for the current data extraction was not 'middle'

AND

- the user-defined data contained results for solid elements

AND

- no explicit 'Layer' id was defined in the file for the solid data.

D3PLOT attempted to set the current layer, some value > 1, as being current for the solid data, and this caused a problem for single integration point solids where only a single 'data slot' was available.

Now fixed. The layer for all element types now defaults to the lesser of 'currently requested' and 'what is legal for this element type'.

- **Case 20449**

When a model contains a binout file it is possible to extract data for 'other' item types, which includes database cross-sections.

This works for data plotting etc., but unfortunately when a cross-section data component (e.g. xsec_f_x) was used in a 'simple formula' for a user-defined component D3PLOT rejected it saying that it was an invalid type.

- **Case 19657**

When user-defined data components were used as the basis of Criterion (principal stress / strain) plots these were not 'remembered' correctly when writing and rereading settings files.

This would affect both explicit use of settings files (to remember an earlier configuration) and also their implicit use when using Reporter to generate plots. Error messages about component sub-numbers would appear and the plot would not be correct.

- **Case 19426**

Two problems have been found when using User-defined data components:

(1) If a user-defined component was created using an external ASCII file of data and read in, then an attempt to plot something *not* in that file, for example data in a shell at an integration point which has not had data supplied, could result in a crash.

This can be worked around by supplying that data, or not requesting data that has not been supplied.

(2) When using WRITE to display a summary of user-defined tensor data if the LOCAL frame of reference was requested the transformation from global to local was not applied, and the printed results were still be in the global system. There is no workaround for this.

- **Case 20261**

When D3PLOT is reading a 'rigid compressed' database, in which LS-DYNA has omitted results for rigid shell elements (DCOMP on card

*DATABASE_EXTENT_BINARY) then errors could occur when processing user-defined data for shells.

- **Case 19549**

When creating user-defined data components for beams the internal logic did not consider the possibility that beams could have data at multiple integration points.

For the 'basic' force and moment data, and also for the 'extra' resultant plastic data, this is correct. However when 'Integrated' beams write out extra data at integration points it becomes necessary to consider the integration point for user-defined data, and this was not being done. The effect was that data was computed and stored for whichever integration point was current, and then not updated if the integration point was changed.

- **Case 21275**

In the help text describing the nomenclature to be used when referring to solid and shell 'extra' data in simple formula data components the syntax is wrong. It states that

+ Extra component n for solids should be SOX_n

+ User defined components for, say, nodal scalar data component n should be UNOS_n

This is incorrect: there should not be any underscore ('_') character, and the correct syntax is SOXn, UNOSn, etc.

This has been corrected. (The user manual has always contained the correct syntax, it is only the 'help' texts that were wrong.)

Utilities

- **Case 20886**

In the PTF COMPRESS menu if the max family size was set to a large number (> 8388607KB) each state would get written to a separate PTF file, whereas one would expect that they would all get written to one file. The max family size is now limited to a maximum of 8388607KB.

- **Case 18107**

Rigid data compression was not being carried through into D3PLOTs cut down PTF files, meaning if the user selected all parts and data components D3PLOTs 'compressed' files would be larger than the originals for a given state.

If the original model has rigid data compression this is now carried through into D3PLOTs cut down PTF files.

- **Case 21418**

If the PTF cut down window was open and the Blank menu was open in the docked area, clicking on Select Parts would incorrectly display the selection menu from the Blank menu. This has been fixed in V11.

Views

- **Case 20933**

If you had two windows, M1 in W1, but M1 + M2 in W2, then exporting the view from W1 to all other windows using the '0' shortcut appeared to work. However as soon as you did any dynamic viewing in W2 it jumped back to its original view.

This was because the internal export of viewing data from W1 to W2 was only being applied properly to the first model in that window, with the 2nd and subsequent ones getting only a partial update. This became apparent as soon as they were redrawn.

Fixed by making the view update apply correctly to all models in a window.

Write

- **Case 20601**

In the WRITE menu the Current Vector and Velocity Vector components did not output any data for airbag particles. This has been fixed.

- **Case 20262**

When using WRITE to list data nothing was reported for the 2D max/min principal stresses and strains for shells. This is now fixed.

XY Plot

- **Case 20152**

If a model had solid elements with 8 integration points and NINTSLD was set to 8 on the DATABASE_EXTENT_BINARY card, the values in an XY plot did not match up with the contoured values.

The contoured values were correct as they averaged the results from the 8 integration points, but the XY plot failed to do this and was plotting the results for the first integration point.

When doing an XY plot the values at the 8 integration points are now averaged.

ZTF

- **Case 20871**

If you read in a model without a ZTF file, then created one and went to File->Reread model, the ZTF file did not get read.

- **Case 20882**

D3PLOT was labelling joints with their internal number even if they had been defined with *CONSTRAINED_JOINT_ID. If a ZTF file is present it will now use the label defined on this card.

2.3   T/HIS

2.3.1   Bugs Fixed in 11

Capture

- **Case 21745**

FAST-TCF script sent to REPORTER didn't contain the comment line at the top containing the image information.

When using T/HIS and REPORTER together if the image format was set to JPEG then the image information wasn't returned to REPORTER correctly when a new T/HIS object was created in REPORTER.

This has been corrected in version 11.

The default image type in T/HIS to has also been changed from JPEG to 24bit PNG as one pixel thick lines don't show up very well in JPEG files. There is a preference option that a user can use to change the default back to JPEG if they want to. As far as FAST-TCF is concerned it won't make a difference for existing scripts as the FAST-TCF 'image' command actually specifies which format to use.

Data access

- **Case 20150**

Some models could show very small values of PEMAG (plastic strain magnitude) when there was no effective plastic strain.

This has been fixed. If there is no effective plastic strain PEMAG will be zero also.

Fasttcf

- **Case 21720**

When running a FAST-TCF script RUN_DIR1 is now the directory of the model that is mapped to model 1 in the FAST-TCF script. Previously RUN_DIR1 would be the directory of the 1st model loaded into T/HIS.

For example if T/HIS has read 2 models from directories

/test/model1/...
/test/model2/...

then if a FAST-TCF script uses $RUN_DIR1 and model 1 in the FAST-TCF script is mapped to the model in /test/model1 then $RUN_DIR1 would be /test/model1.

If the same script was then run with the model 1 mapped to the model in /test/model2 then $RUN_DIR1 would be /test/model2

- **Case 20585**

The option to align the zero value in the 2nd y axis with that in the main y axis wasn't supported by FAST-TCF. This has now been added.

/setup y2_align on   turns on the option to align the axes

/setup y2_align off   turns off the option to align the axes

- **Case 20808**

T/HIS could crash is a FAST-TCF script containing very long lines was played back.

This could show up if a large number of curves were displayed on a single graph as T/HIS would have written out a single 'display' line containing the complete list of curves.

Previously it was not possible to split the curves across multiple 'display" commands

display curve_1 curve_2 curve_3 ….

display curve_101 curve_102 curve_103 ….

as the 1st thing "display" did was to blank all curves before it unblanked the ones that were wanted. This meant that the 2nd "display" command would blank the curves that the 1st one had unblanked.

In version 11 a "display_add" command has been added that does the same as "display" except it doesn't blank all the curves first. T/HIS has also been modified so that when it writes a FAST-TCF script it now only writes about 256 characters to the "display" line and then adds as many "display_add" lines as needed to complete the list of curves.

- **Case 20290**

T/HIS crashed if the first line in a FAST-TCF script contained an uninitialized variable. This has been fixed.

- **Case 21464**

The XY page layout option in FAST-TCF scripts

e.g layout page 1 XY 2 3

was processed incorrectly. If the script had been written out by T/HIS and hadn't been modified this didn't matter as the script would also contain all the commands to individually position each graph on the page.

Graphics

- **Case 21038**

If the Y2 axis values were set so that the minimum was greater than the maximum and the option to display the Y2 axis wasn't selected then T/HIS could crash.

Preferences

- **Case 20980**

If a users preference file set any of the following preferences to 0.0

this*line_width:

this*grid_width:

this*axis_width:

this*border_width:

then FAST-TCF scripts would be written incorrectly as a value of 0 would be written out using the 'heavy' line style (8 pixels thick).

The 'oa_pref.xml' has also been modified to limit the allowable values for the preferences to 1.0, 2.0, 4.0 and 8.0.

- **Case 21458**

If a FAST-TCF script was recorded with colours for curves set to 'dark_grey' or 'medium_grey' then the correct colours would not be applied to the curves when the script was replayed.

Menus

- **Case 21731**

If a user used the SEARCH/REPLACE option in the STYLE menu to change curve labels then after the operation had completed the search string would get reset but the menu would not be updated. If the user then tried to carry out a second search/replace without changing the search string then nothing would happen as the search string was actually empty.

Version 11 has been modified so that the search string isn't reset after a search/replace operation.

- **Case 21721**

If a group of curves was selected in the WRITE menu then number of curves selected was not displayed correctly (it was always zero). The correct curves were selected and they would be written out, it was just the number of selected curves that was displayed that was wrong.

If you clicked on the red header bar in the curve selection menu where it says 'CURVE GROUP LIST' T/HIS could crash.

Operations

- **Case 21380**

If a curve defined with the x-axis in ms was filtered using the FIR filter option then the output curve could end up with the number of points having been increased by a factor of 1000 and it appearing as though the filter had not been applied.

If the curve was manually converted to seconds, then filtered and then converted back again it would have been correct.

Read

- **Case 9407**

T/HIS has been modified so that if a ZTF file is present T/HIS should now be able to correctly work out which integration point is the top, middle and bottom surfaces when MAXINT has been set to a non default value and shells or thick shells with different numbers of integration points have been output to the THF file.

If an attempt is made to read data for an integration point that doesn't exist for an element T/HIS will now produce an error message and produce a empty output curve.

- **Case 20293**

T/HIS would fail to read a TPRINT ASCII file if the option to output the additional top and bottom surface temperatures was used.

Version 11 will now read the file and extract the Temperature and Flux values along with the 2 additional top and bottom temperature values.

- **Case 20900**

When T/HIS is reading data from files produced by LS-DYNA it is possible for the files to contain invalid numbers if the analysis has gone wrong. Previously T/HIS would read and store these numbers but would then have problems plotting them (in the curve editor they would appear as 'NAN'). T/HIS now detects these invalid values and replaces them automatically with 0.0

- **Case 20703**

If a model with a full pathname longer than 132 characters was read into T/HIS then T/HIS would crash if the directory contained an ASCII GLSTAT file. If the file was removed/renamed then everything would be OK. This problem only applied to the GLSTAT file.

From version 11 onwards the full pathname for the GLSTAT file can be up to 256 characters (the same as all the other files)

- **Case 20537**

In version 11 of T/HIS some error messages have been reclassified as warnings.

Previously if you asked for something like airbag output but didn't have any airbags in your model then LS-DYNA would generate an empty ABSTAT file and T/HIS would generate an error message when it scanned the file. These are now warnings instead of errors as they are safe to ignore.

- **Case 19546**

If a model was run which had some contacts defined that did not contain any master or slave segments then the data written to the LSDA (binout) file for contact forces was inconsistent. T/HIS failed to recognise that some contacts

didn't have any segments and the list of contacts displayed would include some contacts twice.

- **Case 20306**

In version 10.0, 10.1 and 10.2 of T/HIS the thermal flux magnitude at a node was not calculated correctly from the x,y and z components.

If the individual components were read in then the VECTOR curve operation could be used to calculate the magnitue.

- **Case 19566**

If a CSV file was read into T/HIS which had commas at the end of each line the following error message would be incorrectly displayed.

Error in C_ADD_HISTORY_DATA: attempt to add history to a curve that does not exist (11)

- **Case 20169**

Version 10 of T/HIS would have generated an invalid warning message if a modal with nodal forces contained the same node ID in more than one nodal force group. In the D3PLOT -> T/HIS link this warning message could have caused a crash on Windows.

- **Case 20903**

If the number of data values written out to the THF (d3thdt) file for a Shell or Thick Shell was greater than 300 T/HIS would read and plot Shell Forces and Strain values incorrectly. This could occur if either the number of integration points specified for the element was very large or if a large number of extra time history variables had been requested.

- **Case 21489**

If whole model stonewall energy was plotted then T/HIS would actually extract and plot the stonewall energy from the 1st stonewall in the model. If the model only had one stonewall this wouldn't matter but if it had multiple stonewalls then the values would be wrong.

- **Case 21446**

T/HIS would not read the ASCII SECFORC file produced by the SMP version of R6.1.1.

T/his link

- **Case 17919**

In version 11 the D3PLOT->T/HIS link 'Timeline' option has been modified so that integrates better with QuickPick.

'Timeline' is now one of the QuickPick modes that can be selected when running the D3PLOT->T/HIS link. If the timeline isn't on then setting QuickPick to 'Timeline' will turn on the timeline and make clicking in a graph move the timeline.

If the QuickPick mode is then changed to something else the timeline remains on but clicking in the graph then activates the normal QuickPick function and doesn't move the timeline. This means it is now possible to use most QuickPick functions while the timeline is on.

If QuickPick is set to 'Timeline' and the timeline is turned off then QuickPick is reset to the mode that was active before Timeline was selected.

2.4   REPORTER

2.4.1   Bugs Fixed in 11

Errortrapping

- **Case 14872**

Reporter will now give a error and offer to show you the log file if it cannot read a file from the command line. Previously nothing happened and it was difficult to see what was going wrong.

Graphics

- **Case 18206**

You could not set a border around image file objects. This has been corrected.

Hyperlink

- **Case 21176**

Hyperlinks did not work correctly for textbox objects. Now fixed.

- **Case 21172**

Hyperlinks were not written correctly in pdf files. Now fixed.

Manual

- **Case 17296**

You could only access the manual on linux if the NETSTART environment variable was set. Reporter now searches for a web browser automatically.

Output

- **Case 21245**

The library program which reported the LS-DYNA version and revision from the otf file did not work correctly for new (R7) LS-DYNA output because there is now a new 'SVN Version' line in the otf file.
Additionally the version and revision were expected to be to be a single 'word'. Now fixed.

- **Case 21242**

Outlines were not written for D3PLOT/T/HIS or File Image type objects to PowerPoints. Now added.

- **Case 17920**

When writing a pdf file jpeg images are now written as jpegs rather than pngs as they can be much smaller.

Read

- **Case 21235**

Reporter would hang when reading a template file if one of the page titles in the file contained an ampersand (&). This was because the ampersand was not escaped properly when writing the template.

Scripting

- **Case 7769**

Reporter could think that a script had changed when cancelling from the editor if the script was created on windows but edited on unix. Now fixed.

User interface

- **Case 19071**

When editing objects on the master page windows could get hidden by the main reporter window instead of staying on the top. Now fixed.

Variables

- **Case 21346**

REPORTER could crash if you created a table that used a library program for a cell and you saved the output of the program to a variable. Now fixed.

2.5 Shell

2.5.1 Bugs fixed in 11

General

- **Case 18577**

The command line options -case and -mcheck can now be specified in the LS-DYNA submission OPTIONs menu.

- **Case 17704**

The shell would not allow more than 4000MWords to be specified for the memory. This has been fixed.

Multiple jobs

- **Case 19196**

The shell now strips leading and trailing spaces from variable names when reading them from the listing file. If they aren't then the command line sent from the shell to start Reporter is invalid and it does not run as expected.

3 Enhancements

The enhancements and bug-fixes for each program have been broken down into a series of topics. Within each topic enhancements and bug-fixes are listed by case ID (most recent first).

3.1 PRIMER

3.1.1 Enhancements in Version 11

Airbag

- **Case 16910**

An editing panel for *AIRBAG_REFERENCE_GEOMETRY keyword is now available. By using this a new *AIRBAG_REFERENCE_GEOMETRY card can be created and an existing *AIRBAG_REFERENCE_GEOMETRY card can be edited.

- **Case 16254**

When the type of airbag is changed in the Airbag editor as much data as possible is now transferred over between old and new definitions.
Each data field in the old definition is inspected, and if a data field of exactly the same name (eg 'C23') and type (eg integer, float, loadcurve, etc) is found in the new definition then the old data value is transferred to the new. (The data fields do not have to be in the same row / column locations.)

- **Case 16118**

Added the ability to create *AIRBAG_REFERENCE_GEOMETRY cards from current nodal coordinates in a model.

- **Case 13465**

*AIRBAG_REFERENCE_GEOMETRY can now be edited in PRIMER.

Airbag folding

- **Case 18417**

When starting the airbag folder nodal coordinates are now automatically switched back to the 'normal' node coordinates if the user was viewing reference coordinates.

- **Case 15125**

If sets and/or layers are used for a fold the colour of the 'Sets and layers' button is now changed to be green.

Assembly

- **Case 17311**

Model build does not renumber parts, materials or sections, so in the event of a label clash model build will be terminated.
This is now explained more clearly in an information panel for interactive build and in the messages sent to <stdout>.

- **Case 21369**

Pedestrian head model build now gives target coordinates, contact point coordinates and JNCAP (aim point) coordinates in the info file.

Auto create

- **Case 11383**

The *CONSTRAINED_EXTRA_NODES editing panel now provides 'Auto-create' functionality.

Belts

- **Case 18741**

The seatbelt fitter has been extended so that the twist of the belt at end and intermediate fixed points (eg sliprings) can now be defined explicitly by giving two extra 'twist' nodes on the path point definition line.
If present these define the vector that the belt path makes at that point, providing an alternative to lining the belt up by eye using the twist options in the belt path editor.
Note that these twist nodes define a vector but **not** a location in space: the belt centreline location at that point is still defined by a path node or explicit coordinate.

- **Case 18283**

When using the seatbelt fitter there is an option to create contact between the belt and the dummy 'structure'.
Historically the 'structure' (master) side of this contact has been a segment set, since the way dummy structure is defined permits an arbitrary mixture of elements and element types, and this was the simplest way to collect them together.
However users have requested that this should be a part set instead since this would make it easier to create 'generic' belted dummies. Therefore the following logic is used:

1) If the part(s) that contain elements making up the structure all have 100% of the elements within them selected (which will usually be the case) then a part set will be used to define the contact.

2) If this is not the case, ie if only some elements in one or more parts have been used, them the original logic of a segment set will be used.

- **Case 17699**

A problem with fitting of 2D seatbelts (shells or 2D belt elements) was that where the belt path passed through concave areas of dummy mesh (eg thigh to lower torso region) it would tend to follow the dummy shape and 'crease' the belt mesh into these regions.
This wasn't wrong in itself, but it could result in the belt becoming too creased during the analysis and giving unrealistic results.
Therefore an optional limiting transverse curvature angle has been added to the belt fitter. If given a non-zero value this limits the angular difference between the outward normal (radial) vectors of transversely adjacent belt segments.

When set to a sensible value, typically 10 to 20 degrees, this limits the tendency of a multi-row belt to 'crease' into hollows in the dummy mesh, instead it will bridge across these hollows to some extent.

- **Case 17698**

In seatbelt fitting it has been possible to control the radial vector of the belt at any path point by adjusting 'twist handles', but the transverse vector (rotation about radial) has not been adjustable.
Transverse adjustment is now possible as follows:

(1) Left mouse click on a twist handle rotates the radial vector as before.
(2) Middle mouse click on a twist handle rotates the transverse vector.

There are some limitations to this: the radial vector can be rotated at any point, but the transverse vector can only be rotated at end, retractor, slipring and acute angle points. The transverse angle at intermediate points is still controlled internally by the belt fitter.
(See also cases 13735, 15110, 16661, 16839 and 17057)

- **Case 17057**

During belt fitting it is possible to control the twist of the basic belt path by moving 'handles' on that path. However if display of the basic path has been turned off these are not drawn making it impossible to use.
Fixed by checking that the basic path is displayed when entering 'control twist' mode, and turning it on if it isn't.

- **Cases 17694 and 16837**

All the geometrical and fitting parameters used during seatbelt fitting can now be set via 'oa_pref' preferences, under category 'seatbelt_fitting'.

- **Case 16791**

When performing manual seatbelt fitting the 'meshing' process requires the user to define the parts to be used for the belt elements if these have not already been specified. Historically the property selection panel offered a selection of all parts in the model, regardless of their element types.
This has now been modified so that only existing parts that use *MAT_SEATBELT or one of the FABRIC material options appear in the list, saving the need for the user to perform manual filtering prior to selection.

- **Case 16249**

When a seatbelt is meshed the centreline length of each section is now reported as it is generated.

- **Cases 17696, 16841 and 16065**

It is now possible to create *DATABASE_CROSS_SECTION definitions in a seatbelt definition, and to 'remember' these across subsequent remeshes. The process works as follows:

(1) Cross-sections can only be defined at belt path points, so it may be necessary to create extra points on the belt path in order to locate these.

In the belt path editor the existing popup which defines attributes of points now has an option 'Cross section' which flags that point as having a section defined. This can be done at any point including end and fixed points.

(2) When the belt is meshed after fitting a cross section is created at the relevant path point locations on the belt mesh.
In order to give well-conditioned cuts the actual cut location will be in the middle of the element immediately after the cut point, unless the point is at the end of a section where the cut will be in the middle of the preceding element.
If a section is defined at a fixed point between two belt path segments, typically at a slipring location, two sections will be defined: one 1/2 an element 'before' it and the other 1/2 an element 'after' it.

(3) Cut sections use the _PLANE suffix, and are located on the belt centreline with their outward normal axis pointing along the line of the belt. They are given a diameter equal to 1.1x the belt width so as to try to minimise the chances of including unwanted elements. The part set used will contain only the part used for the elements at the cut location.

(4) Section locations are 'remembered' in the *BELT keywords written out after *END, so subsequent refitting (either manual or automatic) will detect and recreate them.

For this reason sections created during belt fitting are 'owned' by the belt definition, and this ownership will lock them against deletion.

Checking

- **Case 19848**

PRIMER model checking function now has user control for selection of a sub-set of categories and of includes to which the check will be applied. If the model is written out, the include check status will be encoded on a $PR special comment.

- **Case 19818**

Selective model checking has been added. The user may now exclude categories of items and/or include files from checking.

- **Case 19581**

The drop-down off RIGIDWALL on the error check tree now has an option to write error nodes, such as those found behind the wall, to a node set.

- **Case 19109**

Improved the message given after autofixing cohesive elements that are not correctly attached to shells to tell the user if any could not be fixed due to complex geometry.

- **Case 18367**

LS-DYNA will not initialize if both a rigid part and nodal rigid body exist at the same label and this is referred to by BOUNDARY_PRESCRIBED_MOTION. So the warning has been promoted to an error.

- **Case 18191**

Added an optional part size check to the rigid/deformable continuity checks - allows the user to specify a maximum part size where a error is only reported if the discontinuous part is greater than that size.

- **Case 18063**

PRIMER now checks for loadcurve errors in mat_low_density_foam.

- **Case 17220**

Added error checks for MAT28 (RESULTANT PLASTICITY).

- **Case 16610**

Error check for field LCA in MAT_126 is added. According to this if LCA is not installed then an error message is popped up.

- **Case 16380**

User defined checks can now be written in JavaScript

- **Case 16303**

The duplicate element check now has a switch will enable elements to be reported as duplicates even if they are off different parts.

- **Case 16210**

Added element quality check jacobian.

- **Case 15996**

New optional part checks have been added for normal consistency and cracks within parts.

- **Case 14597**

A warning is displayed if some of the nodes on a shell are used in *AIRBAG_REFERENCE_GEOMETRY, while rest of the nodes in that particular shell are not used in AIRBAG_REFERENCE_GEOMETRY.

- **Case 8823**

The check for duplicated elements may now be configured to consider overlapping elements in different parts as duplicates, previously the elements needed to be in the same part.

Clipboard

- **Case 12197**

Empty Clip Board and -SUB buttons are now greyed out until clipboard is populated.

- **Case 16378**

'Create sub-model from parts' has been added to the options on Clipboard > save model. This function uses a recursive lookup of cross references through part and node sets as necessary to attempt to create a representative sub-model from an initial selection of parts. It provides a more comprehensive treatment than 'Find referenced items' and contains special logic to manage Primer connections.

Coating

- **Case 16987**

Coat tool provides 'Undo' button to reject shell/segment creation

Composites

- **Case 20852**

PRIMER will not write layer data for PART_COMPOSITE cards with one layer per line rather than 2 layers per line - this is easier to read in the deck.

Connections

- **Case 19683**

Connection bolts which use zero length discrete beams have been modified to use *ELEMENT_BEAM_THICKNESS which each refer to a *DEFINE_COORDINATE_NODES to define the initial axis of the beam. The values of and will be taken from the applicable section card.

- **Case 18634**

When a connection bolt is made which uses a discrete beam, ELEMENT_BEAM_THICKNESS is now used and a coordinate system created.

- **Case 18600**

The feature angle can now be set via the connections option panel and by a preference. This aids in creating MIG type spotwelds along a feature line.

- **Case 18164**

A 'set columns' button has been added to the connection table which will allow the user to set up the appropriate subset of available view options for the currently displayed set of connections which may have been filtered interactively

- **Case 17614**

*MAT_SPR is now a valid material type for solid spotweld connection types.

- **Case 17474**

MAT_SPR_JLR can now be used as a solid spotweld connection part within PRIMERs connection features.

- **Case 17307**

Connection type filters have been added to connection selection in the connection panel.

Contacts

- **Case 20677**

The JavaScript function Contact.Interactions() has been created to replace the deprecated FindInteractions(). It returns an array of objects with properties node, shell, s, t parametrics, penetration value and segment thickness value

- **Case 18639**

The initial settings for the SPR and MPR fields when creating contacts can now be controled via preferences.

- **Case 16844**

There is a new option to consider visible entities when using the Contacts->Only feature.

Contour

- **Case 17646**

You can now contour plot element formulation for ALE2D type shells.

- **Case 16262**

Yield stress contour plotting is made available for MAT_99.

Control

- **Case 20324**

The generic *CONTROL editor now includes 'Edit' links to stand-alone editing panels for open-ended cards. Additionally, PRIMER issues a warning if an open-ended card is simply toggled on in the generic editor without specifying any data

- **Case 19068**

'Convert to implicit' now has more control over which materials are converted to elastic.

Cut section

- **Cases 8759 and 17317**

*DATABASE_CROSS_SECTION_PLANE section properties can now be calculated and displayed in the same way as 'native' PRIMER cut sections. (_SET definitions cannot be displayed in this way)
The cross-section editing panel now has a Properties button which launches the cut-section properties panel for that section. The keyword editor for cross-sections also has a properties button on its 'options' popup menu which does the same thing.
Properties are based on the geometry of the cut section, taking into account radius, lenl and lenm values, and also any part set psid that is defined.
Note: Blanking is also considered. Blanked elements will not be included in the cross-section calculation.

Database

- **Case 21037**

The calculated size when creating a database cross section definition in PRIMER is now a bigger, to prevent long, thin sections on thin models, and sections which do not capture all the visible parts.

- **Case 19501**

The DATABASE_ASCII panel has had an extensive redesign and simplification.

Deleting

- **Case 18619**

Added the ability to fire off a keyword editor from the 'why wasn't I deleted?' panel that contains all locked items.

Dummies

- **Case 19773**

Users of encrypted models (typically crash dummies) from 3rd party suppliers face the problem that some data necessary for the input deck to be consistent may be 'hidden' inside the encrypted blocks meaning that PRIMER will generate errors and warnings during pre-processing.
Various solutions for this exist, one of which is for the supplier also to provide an 'alternate' file containing a simplified version of the missing data which can be included during pre-processing.
This works, but it creates a new problem which is that the input deck must be edited prior to analysis to replace the 'alternate' include file with the 'encrypted' one, and this presents work flow problems.

PRIMER V11 provides two solutions to this problem:

(1) If only Material and Loadcurve data are present in the alternate file then these can be placed in an *ENCRYPTED block after *END where PRIMER will use them to resolve the missing items.
This is the preferred solution since PRIMER 'knows' that these are encrypted definitions that must not be changed, and implements special logic that prevents them from being deleted or relabelled.

(2) For cases where items other than Materials and Loadcurves are provided in this way it is now possible to use the *INCLUDE statement after *END.
PRIMER will read the include file in the usual way, and its contents will be treated as 'normal' in every sense, however LS-DYNA will not see this file since it is after *END. (The encrypted include file can be included in the main deck, where it will be remembered but not processed by PRIMER.)
This satisifes the requirement that the alternate definitions be visible only during pre-processing, without any need to edit the input deck prior to subsequent analysis. However there are some limitations:

(a) PRIMER does not 'know' that the items within this include file are alternate definitions which must not be relabelled or deleted, so the checks provided in method (1) above are not active.
(b) There are some limitations upon what this include file may contain, and in particular:

- It may not contain a child include file.
- It may not contain *CASE statements
- Any information after *END in the file will be ignored.

Despite these limitations this should still be a useful facility in these cases.

Element quality

- **Case 17337**

Element quality optimisation - accessible via Keywords->Node->Drag - function has been rewritten and should now give faster results

Geometry

- **Case 16371**

The Node menu now includes an option to create nodes at geometric point locations

Graphics

- **Case 18152**

Graphics in PRIMER have been accelerated, with typical speed-ups in the range 4x to 10x being reported on modern hardware.
'Old style' graphics are still supported, and PRIMER will revert automatically to these on platforms that do not support the new features. The user can also 'tune' graphics manually if the new features cause problems.

- **Cases 4076, 11788, 11901 and 18123**

From version 11 onwards beam end releases can be drawn and labelled using the Entity menu. A circular blob is displayed 10% of the way along the beam if the end releases are defined at node 1 and 90% of the way along if they are at node 2.

- **Case 17309**

Object menus used for selection now have a [<<] button at their top right, (replacing the iconise button), that allows them to be 'undocked' from their parent window.
Undocking a menu makes it an autonomous child of the main PRIMER window, so its width and height will no longer be constrained by the boundaries of its original 'parent' box. This means that long titles on menu rows will now be visible in their entirety.
Undocked menus can be redocked at any time using [>>], and they remain logical children of their parent window so that if the parent window is dismissed they too will be dismissed.
(Note that the border between the Right Hand Side (RHS) menu area and the graphics window can now be moved via a drag handle, potentially allowing more width for object menus in their docked position.)

- **Case 17308**

Covered by cases 17106 (resize handle on RHS) and 17309 (undocking of object menus)

- **Cases 2976, 8191, 14604, 16738 and 17036**

In version 11.0 the graticule option has been modified to give the same options as D3PLOT. The graticule can now be switched beteen a 3D mode where the size and location of individual planes in the global X,Y and Z directions can be specified and the default 2D mode. In both modes the grid spacing can either be calculated automatically or a user defined spacing can be set.

- **Cases 4554, 6153 and 9872**

From version 11.0 onwards the following restaint types can now be contoured.

NODAL SPC's
RIGID BODY RESTRAINTS
NODAL RIGID BODY RESTRAINTS

Each restraint direction is shown using a different colour as follows

Grey - None
Red - X
Green - Y
Blue - Z
Yellow - XY
Cyan - YZ
Magenta - ZX
Black - XYZ

Global restaints are drawn using round symbols, while local coordinate system restraints are drawn using triangles.

- **Case 8432**

From version 11.0 onwards nodal SPC codes can now be contoured. Each code is drawn in a different colour using the following key

X - Red
Y - Green
Z - Blue
XY - Yellow
YZ - Cyan
ZX - Magenta
XYZ - White

Translational and Rotational codes can be contoured separately, and different symbols are used to show the difference betwen SPC's defined using the global (circular) or local (triangular) coordinate systems.

Include

- **Case 18216**

If a subset of component includes for a model are scanned and the model is written out, PRIMER will no longer write empty files for the unscanned includes. Also reference to them will be omitted from the master file.

- **Case 18193**

An 'only' option has been added to the include tree popup options menu to blank everything except the selected include file.

- **Case 7520**

You can now temporarily suspend the transformation part of *INCLUDE_TRANSFORM (i.e. you can see them in the original position).

Keyword

- **Case 19074**

Multiple part selection option has been added for DEF_TO_RIG cards.

- **Case 18528**

Parameter names can now be 9 characters long.

- **Case 18096**

A button has been added to the boundary prescribed final geometry card to allow nodes to be removed easily.

- **Case 16836**

Enhancement: Added ONLY option for DEF_TO_RIG card.

- **Case 16835**

Multiple part selection option has been added for DEF_TO_RIG cards.

- **Case 13778**

New editing panels for DATABASE_EXTENT_:

1) AVS
2) MOVIE
3) MPGS

are made available.

- **Case 8826**

Multiple part selection option has been added to DEF_TO_RIG cards.

Keyword editor

- **Case 16593**

Some keywords in PRIMER only have a keyword editor, and no 'scalar' editing panel. Historically this has meant that when such items need to be referred to from other editing panels there is no CREATE or EDIT functionality in the daisy-chain popup menu invoked by a right clock.
This has now been changed so that if no scalar editing panel exists but there **is** a keyword editor option, then invoking CREATE or EDIT in this context will map the keyword editor instead.
In the CREATE case this will initially map showing no items of the required type (even if some exist) and the user will need to create one. This becomes the 'target' item for the data slot in the parent panel.
In the EDIT case only the item referred to in the parent panel will be shown.

Labels

- **Case 20396**

On keyout PRIMER now gives the user the opportunity to fully declash the elements, sets and materials. The same function is available from 'declash labels' under the model renumber tab. The test can be switched off by a preference.

- **Cases 9871 and 10349**

From version 11 onwards the 'Entity' menu now contains an option for labelling PARTS. This option adds a single label at the centre of each PART.

Loadcurves

- **Case 20986**

Added ability to specify which columns are X/Y when importing CSV/TXT data to the loadcurve edit panel.

- **Case 17089**

A new 'curve compare' feature has been added to PRIMER to allow users to plot, compare, and edit mutiple curves on one graph.

- **Case 14625**

You can now dynamically zoom/pan the curve(s) in the edit panel for *DEFINE_CURVE/_TABLE.

- **Case 12364**

You can now specify any lines you wish to skip when importing a CSV file to the loadcurve edit panel.

- **Case 6441**

Added the capability to view/compare stress/strain curves for MAT24 and MAT98 materials.

- **Case 6188**

Added capability to plot stress/strain curves for Johnson-Cook material.

- **Case 4058**

You can now drag, delete and insert points interactively to *DEFINE_CURVE definitions in the *DEFINE_CURVE edit panel.

Local axes

- **Cases 4030, 11145, 12668, 13400 and 16409**

Solid/Shell elements can be oriented along seed solid/shell element or along a specified vector.

Mass

- **Case 18972**

Mass properties function can now be used to report inertia about arbitrary point (not just CofG) and in arbitrary axis system.

Materials

- **Case 20127**

To address problems of 'missing data' when materials and loadcurves are partially or fully encrypted in input decks two changes have been made.

(1) For the conventional Humanetics case where the title field of an encrypted material is used to define its label three further optional fields may be inserted into the label string:

RO=xxxx or ro=xxxx defines the density as xxxx
YM=xxxx or ym=xxxx defines the young's modulus as xxxx
NU=xxxx or nu=xxxx defines the poisson's ratio as xxxx

If these fields are used they must not have any embedded white space within them (although they must be separated by white space). Thus the title line:

10 ro=7.85 is legal
12 ro = 7.85 is illegal (spaces between 'ro', '=' and '7.85'

The density value (RO) will be applied to all structural materials used for solids, beams, shells, etc where density is the first valid field in the material definition. Spring materials ignore this and always use a stiffness of 1.0.
YM and/or NU values will only be applied to material types 1, 9 and 20.

(2) For the case where all material and/or loadcurve data is fully encrypted, with no readable definition of any sort in the main deck, there is now an optional 'encrypted data' block following *END. Its format is:

*ENCRYPTED_START

any number of *MAT cards, in any order
any number of *DEFINE_CURVE cards, in any order

*ENCRYPTED_END

You can have multiple such blocks in a given file, or in any include file. These provide definitions of materials and loadcurves that stand in for those that are fully encrypted. They are used in PRIMER to satisfy cross-references and to provide density for model checking, but will be ignored during the actual analysis where the 'true' decrypted definitions will be used.
The *MAT and *DEFINE_CURVE definitions are entirely conventional and should follow normal keyword syntax rules with the following exception:

1) Only the label and density (cols 1 and 2 on card 1) are required for materials.
2) Only the label (col 1 on card 1) is required for loadcurves.

For both materials and loadcurves all trailing data fields and/or lines may be omitted. For example the following is legal:

*ENCRYPTED_START
$
*MAT_ELASTIC_TITLE
This is a fabricated material
10 7.85e3
$
*DEFINE_CURVE
99
$
*ENCRYPTED_END

You can supply fully or partially populated definitions of materials (of any type) or loadcurves if you wish, but if you do then the definitions should contain valid numbers.
If no data are supplied for materials then all data fields beyond label and density will be zero.
If no data are supplied for a loadcurve then all factors (row 1) will be zero, and two points will be supplied preset to (0,0) (1,1).

- **Cases 4055, 16255 and 19314**

When using the material editor in PRIMER it has historically been the case that when the type of a material is changed (eg from Elastic to Rigid) all existing data is lost, even if data fields in the new definition match those in the old.
This behaviour has been improved so that when material types are changed as much data as possible is carried over from old to new definition. For each data field in the old definition a search is made for a matching field in the new definition, and if it is found then the data is transferred.

In this context a 'matching' data field requires that:

1) The field names (eg 'PR') be identical
2) The data types (eg integer, float, loadcurve) be identical

It is **not** necessary that old and new data fields should lie in the same row / column position.
(This 'old to new' transfer has also been added for Thermal materials and Equations of state)

- **Case 17300**

The TSRFAC field on *MAT_34 can now be a DEFINE_CURVE if greater than +1.0.

Measure

- **Case 15392**

You can now copy/paste the results of Measure.

Mechanism

- **Case 14183 and 16522**

A new 'automake' capability has been added to Mechanism, which will create a mechanism automatically from the structure and connecting elements in the LS-DYNA model.
'Structure', which becomes mechanism assemblies, is defined by structural parts (solids, shells, beams, thick shells) and selected *Constrained items that can be used to connect these parts. It is also possible to grow structure across tied contacts.
'Connections' are derived from *Constrained Joint definitions, and also from short discrete elements (*Element Discrete, and *Element beam using element formulation 6).
The user selects a single 'seed' node anywhere on an eligible piece of structure, and the process then grows outwards from that node building assemblies and connections until it runs out of eligible items.
The user can control the process by selecting whether or not whole keyword categories should be used, or by (de)selecting individual items within those categories. The critical length for 'short' discrete elements can also be controlled.
The process is interactive, taking only a few seconds, and visual feedback shows the mechanism growing assembly by assembly. Settings can be changed at will and the process repeated until a satisfactory solution is obtained, and the user can perform a trial positioning of the mechanism at any time.
As with conventional editing in PRIMER the process always works on a 'scratch' mechanism, which does not affect anything stored in the database. Once a satisfactory result has been obtained it can be stored in the database as a permanent mechanism definition.

- **Cases 11726, 16264, 16522 and 16723**

A new 'stretch' capability has been added to bot Mechanisms and Dummies ot permit interpolation of structure between assemblies when these move.
The user defines a node at each end of a stretch definition, at least one of which must be on an assembly, and also the 'structure' that is to be moved. During dummy and occupant positioning movement of the assemblies will also result in this structure beting 'stretched' by linear interpolation of movement of the two end nodes.
It is also possible to define 3 nodes at either or both ends, which results in an

'encastre' definition capable of interpolating rotation and bending (as in lever arm times rotation) as well as translation.

Menus

- **Case 19819**

Historically Oasys Ltd software has used the convention that a button or text box with a small 'v' has a popup menu associated with it that can be invoked by a right mouse click.
To make it easier for newcomers to the software these popup menus can now also be activated by a left mouse click if the mouse is over the 'v' region. To make it easy to know when this is the case this region is highlighted whenever the mouse moves over it.

- **Case 19815**

Historically the font sizes used in buttons in Oasys Ltd software graphical user interfaces (GUIs) have been allowed to make themselves smaller if the text is too long to fit in the button.
This means that the text string is more likely to be drawn in full, rather than being truncated to fit, but it can also lead to an untidy appearance if a GUI panel contains a mixture of different font sizes.
Generally it is better to adjust the Display Factor to achieve a consistent appearance, but there is now also the option of turning off font scaling in the Options > Menu Attributes panel. This means that all GUI fonts will be the same size, but some buttons may have truncated text strings.
The default is for scaling to be active (on), which is the historical behaviour. As will all Menu Attributes settings Font Scaling can be saved as a preference in the oa_pref file.

Merge

- **Case 10669**

A popup has been added to model merge to allow the user to easily choose different titles for the target model.

Mesh

- **Case 18628**

Mesh=>Extrude=>Shells, can now be used to create thick shells.

- **Case 17872**

MESH-Extrude-Shell, can now be extruded along the shell normal direction.

- **Case 17797**

When creating a hole in the mesh if the hole diameter is made larger than the washer diameter the washer diameter is now automatically increased.

- **Case 16786**

Detach Solid and Beam options are made available under Meshing->Split->Detach. These options will detach the selected solid or beam from rest of the

model, i.e. these selected elements will not share the nodes with unselected elements.

- **Case 16548**

Added an option that sets ESORT=1 on *CONTROL_SOLID, while extruding the shells.

- **Case 14423**

PRIMER can now make a 'cobweb' mesh pattern.

- **Case 11232**

Split operation for pentagonal and tetrahedral elements is implemented.

- **Case 9718**

A Hexa element can now be split in to 8 hexa elements. While performing this operation, the cross references for PART, SET_SOLID, SET_NODE, DATABASE_HISTORY_SOLID, INITIAL_STRAIN_SOLID and INITIAL_STRESS_SOLID on a selected hex elements are also taken care by propagating those to the newly created eight hexa elements. This operation can be performed using Meshing->Split->Multiple->solid panel.

- **Case 9417**

The *DRAG_NODE editing panel now provides a new radio button option 'N1->N2' for node dragging. Selecting 'N1->N2' radio option will allow users to shift source node 'N1' to coordinates of destination node 'N2'.

- **Case 7651**

Solid elements can now be split.

- **Case 6133**

Detach Solid and Beam options are made available under Meshing->Split->Detach. These options will detach the selected solid or beam from rest of the model, i.e. these selected elements will not share the nodes with unselected elements.

Model build

- **Case 20583**

If model build and keyout run from the template > build panel and the build contains connection (xml) files and the option is set to write the master only file, the (default) option of 'use parent layer' for bolt FE is temporarily disabled, so that the FE is written to the master file.

- **Case 19522**

PRIMER now supports a new PEDLEG_UPPER_2 model build type. This is similar to the existing PEDLEG_UPPER type but permits explicit Z coordinate specification for each loadcase.

Nastran

- **Case 19414**

When reading in Nastran models any Hypermesh comments associated with
PARTs, SECTs and MATs are now stored and used as titles on the relevant cards.
Writing out a deck as a Nastran model will now also write the Hypermesh
comments.

Nodal rigid bodies

- **Case 18119**

The *NODAL_RIGID_BODY editing panel now provides a method 'Edge of
hole' to create NRB (NODAL RIGID BODY) around hole. It will allow users to
select any node on the edge of the hole. A node is automatically created at the
centre of the hole and added to a node set. Nodes situated along the circumference
are automatically selected and added to the same set.

Node

- **Case 12564**

Earlier only 'Pick' type node creation method was available if users were to
daisychain into the CREATE NODE panel from elsewhere. Some other valid
single node creation methods have now be made available.

Node transform

- **Case 14101**

You can now temporarily suspend *NODE_TRANSFORM definitions (i.e. you
can see them in the original position).

Orient

- **Case 19675**

A backoff option has been added to orient into contact function.

- **Case 17677**

Command line orient now allows selection of items to be oriented by include file.
PRIMER supports both a selection of the literal contents of the include
(INCL_LITERAL) and a selection of the parts in the include (INCL_PARTS)
which may mean orient propagates to nodes outside the selected include(s).

- **Case 17465**

The 'project to mesh' orient tool now includes an option to back-off by a distance
determined by part thicknesses.

- **Case 15490**

Now user will be able to type/edit the co-ordinates of N1, N2 instead of picking
an existing node. One can translate the object by providing single Node (either N1
or N2), where second node will be origin O (0, 0, 0).

- **Case 8318**

Orient interpolation has gained a 'part only' option. This avoids the problem of topological links across parts causing unexpected orient.

- **Case 4470**

Orient translate, rotate, scale and reflect can now be done in local coordinate system. The user must select an existing *DEFINE_COORDINATE_SYSTEM.

Output

- **Case 18315**

When PRIMER crashes the special crash recovery panel gives the option of performing an 'emergency keyout' of all models. Currently these are written to the user's home directory, or if that cannot be determined then to the current working directory.
We have had a request to be able to define a special 'emergency keyout' directory, so this can now by done via the preference:

primer*emergency_keyout_dir: pathname

If this is defined then it will be used, otherwise the default directory will be used as before.
Note!! No check is made to test whether this directory is valid. The results if it doesn't exist, or is not writeable, are undefined.

Parameter

- **Case 17084**

When using 'Model modified?' to look for changes in a model a problem can arise if a parameter value is changed.
The change to the *PARAMETER definition itself is detected and reported correctly, however anywhere that the parameter is used (ie &name in a data field) will also be reported as a change because the 'modified' function compares the actual underlying data values of fields.
This is not really an error, since it may be useful to know where a change in a parameter value will have an effect, but it can be confusing when there has been no change to the keyword which references the parameter.
Therefore an option to compare parameterised fields by name rather than underlying value has been added. If this is turned on, and a data field in both current and reference models uses the same parameter name, then the data fields will be treated as 'the same' even if the actual value of the parameter in the two models is different.

- **Case 17081**

When integer parameters are used to specify item labels, either explicitly as the items label field, or implicitly by referring to something else (eg a node on an element card), then a problem arises if the model labels are renumbered.
Previously this would simply have resulted in the parameters no longer matching the underlying values in these data fields, so PRIMER would replace the values in these fields with the (correct) integer values and the parameterisation would be

lost.

Following relabelling operations, eg MODEL > RENUMBER, PRIMER now checks all references to integer parameters and presents the user with a list of changed references. It also works out whether all the new references are consistent with one another, and whether any of the old references still remain.

If all references to a parameter have been revised to the new value then it is safe to modify the parameter's value to this value, and the user is given the option to do this.

Therefore it is now possible to maintain parameterisation of input decks where parameters have been used for item labels, so long as this has been done in a consistent way that is compatible with the renumbering process.

Part table

- **Case 20657**

Two new columns have been added to the part table. One to say whether a part is of type COMPOSITE or not, and another to list the number of layers if it is of type COMPOSITE.

Part replace

- **Case 21173**

You now get a warning message if replacing a part in Part->Replace that is referenced by a *INCLUDE_STAMPED_PART definition.

- **Case 18735**

Part replace now allows de-selection of nodes for re-attachment but setting a tolerance or node picking.

- **Case 18166**

The multiple part replace function has been improved so that parts of more than one element type may now be simultaneously imported.

- **Case 14186**

A new function 'replace part assembly' has been added to Part > Replace. This replaces a selection of target parts (typically an assembly) with the contents of a source model. As the import is not restricted to parts/elements, this provides a more versatile way of updating components of a model.

Part tree

- **Case 19491**

Entities are now sketched automatically if the mouse hovers over their entry in the part tree.

- **Case 17124**

Full path hover text can now be displayed for include files in the part tree.

- **Case 16908**

The number of options available when right clicking on an include/model in the part tree when in 'contents' mode has been expanded to match the options available when in normal mode.

Pedestrian

- **Case 19472**

An extra option has been added to the pedestrian markup script to specify a NxM grid of impact points per head impact zone for the EuroNCAP protocol.

- **Case 19428**

The Pedestrian markup script included in the primer_library/scripts folder would sometimes have to ask the user to manually pick intersection points so the script could trim lines. This has been improved so it is less likely to happen.

- **Case 18626**

The pedestrian markup script now has an option to automatically calculate the velocity, angle and energy for each upper leg impact point, as specified by the EuroNCAP protocol.

- **Case 17763**

Pedestrian markup script additions:

- Upper leg impact points can now be created in the bumper area. Previously they could only be created on the bonnet.
- The EuroNCAP grid method has been added as a new protocol.

- **Case 16355**

A 'Build' button has been added to the Pedestrian Markup javascript (in $OA_INSTALL/primer_library/scripts). Pressing it will build a model for each impact point created.

Penetrations

- **Case 18440**

Function '->pencheck.csv' has been added to the contact Penetration check panel. This writes a csv file containing details of part on part interactions for all sliding contacts.

- **Case 15474**

The contact penetration checker has been modified to include affect of settings shledg=1 and soft=2 which mean the penetrating segment is treated with a flat edge rather than a radius.

- **Case 16146**

The contact crossed edge fixer has been enhanced to enable automatic selection of nodes on one side of a crossed edge from a single picked node. (pick & propagate). Also PRIMER now has an automated fixing function (pick, propagate

& fix) for removal of the crossed edge initiated by a single pick of a node on the side to be moved.

Preferences

- **Case 18416**

Added preference for intopt when creating
*ELEMENT_SEATBELT_ACCELEROMETER.

Properties

- **Case 16984**

A 'Save property' capability has been added to PRIMER and D3PLOT, which will capture the current status of the image by storing:

- Attributes of items in model: Colour, display mode, transparency, blanking
- Attributes of the 'entity panel': all switches
- The current view.

Any number of these 'saved properties' can be stored, and the user can toggle back and forth between them.
Properties can also be written to a programme and model-independent properties file, making it possible to copy properties between programmes and models. This can be read in both PRIMER and D3PLOT, and could be read or written by other codes too.

Quick pick

- **Case 16009**

Polygon and circle picking modes added to 'Quick pick'
Historically Quick Pick has only ever worked as 'click to select single item', or 'drag out a rectangle and select all within it'.
This is still the default, but has now been extended so that modes:

+ Polygon pick: define an arbitrary polygon and select all within it.
+ Circle pick: as the default, but dragging is circular.

Have been added. The user can toggle through these modes by clicking on a new button to the top right of the Quick Pick buttons, or by using the 'P' (note upper case) keyboard short-cut.

- **Cases 12171 and 13801**

'Quick pick' in PRIMER now supports polygon selection mode.
The default is still 'click on an item to select it, or drag out a rectangle to select all within', but it is now also possible to toggle through this and two alternative modes:

- Polygon pick: pick three or more points to define an arbitrary screen polygon, and select anything within it.
- Circle pick: like the default rectangle pick, but drags out a circular rather than a

rectangular area.

There is a new picking mode button at the top of the graphics window which controls the current mode, which can also be selected by using the upper case Q short cut key.

Read

- **Case 16396**

A function 'Inc Declash' has been added to the read tab. This can be used to read a set of component includes which may have clashing labels with one another. Each include must be readable in itself. The declash process will attempt to preserve labels as far as possible.

- **Cases 13302 and 12450**

On LINUX the file selector has bene replaced with a new one that allows multiple file filters to be set. This means that is is now possible for the file selector to show all *.ptf* and d3plot* files at the same time.
The new file selector also allows multiple files to be selected and for the view to show more file information.

Renumber

- **Case 21382**

Additions to renumbering/checking for SET COLLECT:

1) The check that your entities are within any relevant numbering ranges does now not apply to SET COLLECT.
2) For the 'Renumber include' panel, SET COLLECT is now ignored, both when checking if they are within range and when renumbering.
3) An option has been added that applies to all other renumbering panels for whether you renumber SET COLLECT or not. The default is to not renumber them.

- **Case 20939**

An option has been added to NOT renumber latent entities when using 'Model renumber' or individual entity renumber panels.

Scripting

- **Case 21282**

Methods AddWidgetItem(), RemoveWidgetItem() and RemoveAllWidgetItems() have been added to the Widget class to help reposition WidgetItems.

- **Case 20894**

It is now possible to read UTF-8 encoded files from scripts using the File class.

- **Case 20602**

The PlayMacro() method now returns whether an error occurred when replaying the macro.

An extra argument has been added that allows the user to control whether the script is terminated if an error occurs. The default is true so existing scripts continue to work in the same way but the user can set it to false if required.

- **Case 20578**

A Part.Mass() method has been added to enable the user to calculate various mass properties for a part.

- **Case 20391**

A Node.Merge() method has been added to JavaScript

- **Case 20125**

A LoadRigidBody class has been added to give access to LS-DYNA *LOAD_RIGID_BODY cards.

- **Case 19733**

Support for *INITIAL_AXIAL_FORCE_BEAM has been added to JavaScript (AxialForceBeam class)

- **Case 19621**

The Model number property is no longer read only.

- **Case 19561**

An optional model number argument can now be given in the Model constructor and Read method.

- **Case 19466**

Several methods have been added to the PrescribedMotion class to enable them to be animated from JavaScript.

- **Case 19464**

Model.ImportInclude() now returns an Include object for the include file imported.

- **Case 19463**

A new background property has been added to the Window class to enable the background colour of windows to be changed.

- **Case 19462**

A new method DeleteInclude() has been added to the Model class to enable an include file to be deleted from JavaScript.

- **Case 19313**

Method RenumberAll() has been added to the Model class.

- **Case 19312**

Method SetFlag() has been added to the Model class.

- **Case 19186**

Checking a script now gives some extra warnings for 'dubious practice'.

- **Case 19072**

New script to create rigid 'spider' in a hole, and to merge spiders when they are in different Include files.

- **Case 18901**

Macro tags can now be given to Widgets in JavaScript with the Widget.macroTag property.

- **Case 18871**

Parameters can now be used for fields in keywords from JavaScript.

- **Case 18750**

A Group class has been added to allow groups to be accessed from JavaScript.

- **Case 18749**

Support for creating/editing _GENERAL sets has been added to JavaScript.

- **Case 18653**

It is now possible to create popups for Widgets in JavaScript

- **Case 18409**

Support for *DATABASE_NODAL_FORCE_GROUP has been added to JavaScript.

- **Case 18262**

Properties left, right, top, bottom, width and height have been added to the Window class.

- **Case 18251**

Window.GetFiles() has been added to allow multiple files to be selected.

- **Case 17962**

JavaScript files are now searched for by extension '*.js*' (previously '*.js')

- **Case 17885**

A new class method Total() has been added to get the total number of items in a model.

- **Case 17657**

Model.Read() can now read Nastran, Radioss, Abaqus and IGES files.

- **Case 17656**

The garbage collection memory size required for a script can now be set by a special comment at the top of the script.

- **Case 17612**

Support for *DATABASE_FORMAT has been added to JavaScript.

- **Case 17545**

If Model.Read() fails it now returns null and the script continues (which is what the documentation says it should do) rather than terminating.

- **Case 17498**

Every JavaScript window now has a X symbol to close the window. Additionally there is now an onClose property for the Window class which is a function to call when the X is clicked.

- **Case 15613**

Added a function to report element area/volume in JavaScript

- **Case 15610 and 15612**

Added a function to give the mass of flagged parts in JavaScript

- **Case 14397**

Rigidify from command line added.

- **Case 13591**

Property 'element_type' has been added to the Part class.

- **Case 13223**

3 new functions have been added to each class to obtain a new label.
FirstFreeLabel(), LastFreeLabel() and NextFreeLabel()
Each takes a model object as argument 1 and an optional include file number for argument 2.
Functions FirstFreeItemLabel(), LastFreeItemLabel() and NextFreeItemLabel() have also been added to the model class. You can specify the type as a string.
e.g. the following are equivalent

Node.FirstFreeLabel(m);
m.FirstFreeItemLabel('NODE');

- **Case 13156**

A new Total() method has been added to keyword classes to easily find out the total number.

- **Case 12626**

Seek() and Tell() functions have been added to the File class.

- **Case 12610**

Window.GetFile() and Window.GetFilename() now have an optional extra argument that can be used to specify the initial directory (for GetFile) or filename (for GetFilename)

- **Case 11005**

GetGeneralData(), RemoveGeneralData() and SetGeneralData() methods added for _GENERAL sets in the Set class.
Set spooling has been improved. Segment sets can now be spooled.
A new 'raw' argument has been added to the Set.StartSpool() method. This allows the 'raw' data to be spooled for _GENERAL, _ADD and _INTERSECT sets rather than 'expanding' the child set data or ranges.

Selection

- **Case 19816**

Selection ('object') menus in PRIMER have historically been constrained to lie within their parent selection panel. They can auto expand to the left or right if necessary, but cannot grow beyond the bounds of that panel.
A new 'auto undock' functionality has been added so that when the mouse is moved into a menu it will automatically undock itself so that it is no longer constrained by its parent window, and will expand as far as possible in order to show more of its contents.
When the mouse is moved back out of the menu it reverts to its original position and parentage.
This can be controlled by Options > Panel behaviour.

- **Cases 17288, 18860 and 19814**

Two enhancements have been made to object menu selection.

(1) The existing 'Filter by string' logic has been extended so that it considers the complete text string in a menu row.
Previously it only considered the item title or description, and omitted the Mnn/TYPElll prefix. Now that the whole menu string is considered the wildcard capability of this filter can be used to select a range by label as well as name.

(2) The 'Keyin' capability where explicit labels can be typed in has been extended to support wildcard syntax in the numeric data.
Two wildcard characters are permitted:

'?' matches a single number.
'*' matches any number of characters

For example:

P3? will select parts 30 to 39
P10* will select parts with any label starting 10...
M*/p10 will select part 10 in all models
M?/P5* will select parts with labels starting 5... in models 1 to 9

Keyin selection is now also subject to whatever filters are currently in force, previously it ignored this.

- **Case 16261**

See cases 17309, 17308 and 17106

- **Case 15428**

Object menus in PRIMER have historically auto-expanded to show content than is wider than their default size, but this expansion is limited to 90% of the width of the box which contains them.
They can now be 'undocked' from this box by clicking on the [<<] button at the top right. (This replaces the iconise button previously in that location.)
When undocked they will become wider and, more importantly, will no longer be children of their parent panel meaning that the user can drag them out to any position and size.
They can be redocked again by using the [>>] button.

Sets

- **Case 17002**

Added ONLY function for SETs.

Spotwelding

- **Case 18497**

A check has been added to report error when a spotweld beam contact is defined by node set on the slave side and *CONTROL_SPOTWELD_BEAM is in use. A model so configured cannot be run in LS-DYNA.

- **Case 16613**

Connections have been modified so that you can now specify a separate part for each layer of solids/beams in a spotweld/adhesive connection type.

- **Case 12104**

When remaking a connection on the connections table, if the connection has no layer information, PRIMER will now populate the layer fields with the information it finds from the spotwelder. The FE was always created.

Translators

- **Case 18731**

A crash in the ABAQUS translator has been fixed.

Units

- **Case 19857**

Updated units for *DEFINE_CPM_GAS_PROPERTIES coefficients.

User interface

- **Case 20607**

A long-standing problem in PRIMER is that floating menu windows can get 'lost behind' the graphics window, or the docked right hand side (RHS) region, and it can be very hard to know how to retrieve them.
Two options have been added to the Options > Panel behaviour menu to help with

this:

(1) An option to keep the RHS docked region lowered.
(2) An option to keep the graphics window lowered.

If used these should eliminate the 'lost behind' problem, although both exhibit behaviour that some users may not like, so they are switchable.
In addition a 'forcible lower' [v] button has been added to the graphics window which, regardless of the settings described above, will forcibly lower both graphics window and all docked windows to the bottom of the stacking order, causing any 'lost' floating menu panels to reappear.

- **Case 19510**

From version 11 onwards the standard windows shortcut keys, CTRL-C, CTRL-X and CTRL V can be used to copy, paste and cut text in text boxes.

- **Case 19166**

It is now possible to use Ctrl+C and Ctrl+V to copy and paste between textboxes on Windows.

- **Case 17106**

The border between the Right Hand Side (RHS) menu area and the graphics window is now no longer fixed.
The default size and layout has not changed, but a resize 'Handle' has been added to the bottom left of the Tools/Keywords window that allows this border to be dragged left and right to change the size of the RHS region. The following functionality is provided:
Left mouse Drags the border, resulting in Tools, Keywords and Tabs region possibly getting a different number of columns, and hence occupying more or less vertical space. The space in the 'docked' region below used for menu panels will shrink or grow vertically and horizontally to use this space.
Middle mouse Returns to the default layout, by default 4 columns of tools and keywords buttons.
Right mouse Maps the Window and Layout panel allowed manual control of RHS region and graphics window layout, and also the ability to set and save these as preferences.

- **Case 13297**

The ability to save the current blanking status of all models as a 'saved blanking state' has been added.
This is done via the 'Save B' button in the View panel, and the user can then use <= and => buttons to cycle through saved blanking states.
There is no limit to the number of blanking states that can be saved, although it quickly becomes difficult to keep track of things if more than a few are used.
There are some further controls in the BLANK panel which allow saved states to be deleted.

Write

- **Case 14944**

A new option has been added to write out the initial geometry of a model as a .ptf (d3plot) file.
The intended purpose of this is to check whether new components clash with existing parts, not only in the initial geometry, but also in the deformed state.
For example it may be proposed to add a rigid lump to the engine of a car. The key file of the lump could be written out as a ptf file and then read into D3Plot along with the existing crash model. By turning on shift deformed on the engine in the crash model, the rigid lump will appear in the correct place on the engine even when the model is in the deformed state. It could then be seen whether any deformed structure would pass through the rigid lump.
IGES geometry can also be written out as PTF files, with the surfaces converted to shells.

Ztf

- **Case 20318**

Output of multiple ZTF files for models containing *CASE has been added.
Each case is treated as a separate model, and gets an individual ZTF file.

3.2 D3PLOT

3.2.1 Enhancements in 11

Blob plots

- **Case 13044**

When external data ('blob') plots are in use the actual maximum and minimum data values in the file are now displayed underneath the contour bar. This makes it possible to visualise the max & min values in the file when a contour bar with user-specified levels is being used.

Also the fact that it possible to specify external data at nodes as well as at [x,y,z] points in space was omitted from early versions of the user manual. this has now been corrected.

Command Line

- **Case 20323**

The dialogue command /OTHER_PLOTTING has been added to select and plot LSDA (binout) components.

Contour

- **Case 18070**

Historically automatic contour bands in D3PLOT have auto-scaled animations by calculating the 'envelope' of max/min values over all frames of the animation, then displaying it scaled to this single set of bands. Therefore all frames of the animation use the same contour scale, and the numeric values of a band of colour remain constant over the whole animation.

A new 'Automatic each frame' contouring mode has been added which instead auto-scales each frame of an animation independently. Therefore the numeric values of a contour band are likely to change from frame to frame. This makes an animation equivalent to a sequence of separately auto-scaled static plots.

The first mode, automatic all frames, remains the default; however the second mode may be useful in cases where data values vary by a large amount over the animation, and the large values effectively 'wipe out' any variation in frames with lower values.

Cut section

- **Case 17580**

The dialogue command /CUT_SECTION WRITE_FORCES has been added to write cut section forces to a CSV file.

- **Case 15653**

When cut-sections are active the current origin coordinate of the plane is now always displayed under the picture showing the plane specification method.

This coordinate may be dragged out for cut and paste operations.

- **Case 20441**

The display of the vector values entered into the cut section panel were sometimes getting truncated unnecessarily. This has been fixed.

- **Case 12617**

D3PLOT can now create an XY plot of DATABASE_CROSS_SECTION force vs. section position. This can be done by selecting COMPOSITE option in XY_DATA and then selecting the LOAD_PATH segments to plot, setting X to be the position of each cross-section and Y the force.

- **Case 16947**

Version 11 of D3PLOT has a new feature called LOAD PATHS that can be used to display the flow of forces through a structure. The Load Paths are defined in PRIMER by selecting the *DATABASE_CROSS_SECTION definitions that form a load path through the structure. D3PLOT can then access these definitions via the ZTF and combine them with the cross section forces written to the LSDA (binout) binary file.

Data access

- **Case 20147**

Historically D3PLOT has plotted nodal temperatures if present in the database, but only as 'scalar temperatures at nodes'.

So thermal analyses in LS-DYNA are capable of producing some or all of:

- Temperatures at nodes for top, middle and bottom shell surfaces.

- Temperature fluxes (X, Y and Z components)

- The rate of change of temperature with time (dTemp/dTime)

These are now all supported by D3PLOT and may be contoured and extracted numerically as nodal quantities. Fluxes may also be plotted in 'vector' mode to show magnitude and direction.

- **Case 20997**

D3PLOT 11 supports the following data components from the DCFAIL part of the LSDA (binout) file.

Failure Function

Normal Failure

Shear Failure

Bending Failure

Area

These are additional data components that can be generated for solid and hex assembly spotwelds.

Version 11 also supports the 'Spotweld Length' data component written to the SWFORC part of the LSDA (binout) file.

Deform

- **Case 13348**

An 'Auto' button has been added to the Deform->Magnify panel to automatically set the magnification factors so that the maximum displacement is a user specified percentage of the model size.

Femzip

- **Case 19459**

Previously the FEMZIP reader, an independent process, has been subject to a 20 second time-out period on initialisation.

This means that if it does not respond to the main D3PLOT process within 20 seconds it is assumed to have died and D3PLOT gives up on it.

This has proved to be too short a time on some systems, so:

+ The default timeout period is now 60 seconds

+ This value is configurable via environment variable D3PLOT_FR_TIMEOUT

Formatting

- **Case 18170**

Scientific and General have been added to the list of available number formats for the contour bar, graticule and external data.

Graphics

- **Case 18738**

In version 11.0 onwards a change has been made and the graphics tuning options in the UTILITIES->GRAPHICS menu have been turned on by default.

- **Case 17729**

When using 'true' sections the display of circular beams could, if the scale was made small enough, result in their being displayed in a shaded plot as a 'thin line with large blobs on its ends'.

This was caused by an error when processing solid circular beams, which have an internal diameter of zero. The external tube was drawn correctly, but the zero internal diameter also resulted in their being drawn as 'section cannot be computed' symbols, which is a line with blobs.

This has been corrected so that solid circular beams are now drawn correctly, and in addition the 'section cannot be computed' symbol has been modified to make the blobs much smaller.

- **Case 18540**

If multiple models are in the same Window and the user has set the colours of the models to make it easier to see them then the model titles are now also drawn using those colours.

- **Case 20343**

If multiple models are in the same Window and the user has set the colours of the models to make it easier to see them then the model titles are now also drawn using those colours.

- **Case 18141**

The grid spacing used in the 2D graticule can now be defined by the user.

- **Case 17548**

The format of the numbers displayed by D3PLOT for Max & Min values can now be controlled by the user from the Max & Min panel.

It can be set to 'AUTO' (default), 'SCIENTIFIC', 'GENERAL' or 'MANUAL' and the number of decimal places and exponent number can be specified.

The following preferences can be used to set the defaults:

d3plot*maxmin_number_format_type

d3plot*maxmin_exponent

d3plot*maxmin_dec_places

Image

- **Case 19203**

On *nix platforms when D3PLOT and PRIMER wrote 2x or 4x scaled image files (to .jpg, .png, .pdf, etc) it would sometimes be the case that the larger fonts would come out the right size but in the wrong typeface. For example something which looked like (but wasn't) Times instead of Helvetica.

This was due to a limitation of the default X11 font installation on *nix platforms, and while it could be solved by downloading and installing scalable 'true type' fonts this is a tricky business even for geeks!

Therefore PRIMER and D3PLOT now detect for themselves when the correct 'large' font is not available and generate it internally by scaling up a smaller typeface of the correct format. This gives a slightly 'pixelated' effect when viewed at full size, but 2x and 4x images are rarely viewed at their full scale so in practice this doesn't matter very much. Certainly it is better than having the wrong font!

Local axes

- **Case 16533**

Local coordinate systems setup in the Deform->Shif def and Deform->Ref node menus can now be saved and reused.

If they have been saved, they will be available for selection in the Measure menu and the Ref frame->User-defined menu. The User-defined frame of reference system is now also displayed in the graphics screen.

Coordinate systems defined by a *DEFINE_COORDINATE keyword will also be available for selection via the ZTF file.

Dialogue commands are available to save and retrieve the coordinate systems, meaning they can be accessed in a JavaScript.

Menus

- **Case 19512**

In version 11.0 the directory selector menu has been replaced with a new one that includes a top bar into which a path can be pasted.

- **Case 18989**

In version 11.0 the directory selector menu has been replaced with a new one that includes a top bar into which a path can be pasted.

Multiple windows

- **Case 12493**

Several users have requested some means of synchronising animation for multiple models that have different output frequencies. For example:

M1: 0.0, 0.1, 0.2, 0.3, etc

M2: 0.0, 0.2, 0.4, 0.6, etc

In the example here it would be possible to match results by omitting every other state from M1, but it is not always the case that times in model A are an integer multiple of times in model B, so this is not a perfect solution.

D3PLOT has always had the ability to interpolate data by time for animation, and this provides a much better mechanism for matching up models with different output frequencies. However this capability is rather hidden away under the ANIM > Custom ... popup menu, so it has been made clearer and easier to use in the following ways:

The (states) ANIM > popup menu now has an INTERPOLATE option which takes you directly to the 'Set times' panel.

This panel allows you to define a [start time] [time interval] pend time] that will be applied to animation in all selected windows. An explanatory note has been added to this panel to make its usage and applicability clearer.

Where the selected window(s) contain(s) multiple models interpolation is applied to all models, so that the affected windows will then animate results at exactly matching time states, regardless of the actual times of the results.

(See also case 13238)

- **Case 13238**

See case 12493

Nastran

- **Case 18139**

D3PLOT can now read results from a Nastran OP2 file. The following solution types are supported:

SOL 101 Linear Statics

SOL 103 Normal Modes

SOL 106 Non-Linear Statics

SOL 107 Direct Complex Eigenvalues

SOL 108 Direct Frequency Response

SOL 109 Direct Transient Response

SOL 110 Modal Complex Eigenvalues

SOL 111 Modal Frequency Response

SOL 112 Modal Transient Response

Preferences

- **Case 20436**

The preference d3plot*platback_method has been added to specify the background movie playback method (STREAMED or CACHED).

- **Case 17533**

The preference d3plot*draw_update has been added to control when the graphics are updated when the state slider is dragged. It can be set to RELEASE or SLIDE.

Read

- **Case 21228**

Version 11 of D3PLOT supports a number of different 'interface force' files that LS-DYNA can now generate. Previously D3PLOT only supported and looked for the CTF (contact force files) associated with a set of PTF files but LS-DYNA can now generate Blast Force Files, FSI Force Files, CPM Force files and DEM Force Files that.

Although D3PLOT can read all of these file types only one of these can be read at any time along with the PTF files. The 'Open Model' menu now contains a popup

menu that allows the user to select which of these additional files is read if they have run an analysis that generates more than one of these files.

- **Case 20902**

D3PLOT can have problems, including crashes, if input decks contain illegal numerical values such as the IEEE 'NaN' (Not A Number) value.

These can be detected and removed by Utilities, Clamp data, but this is not on by default so often the first indication that these values are present in the results files is a crash or freeze.

Testing reveals that the time taken to inspect input data and remove these values, replacing them with zero, is negligible. Therefore from V11 onwards they will be removed by default.

(Utilities, Clamp data is not affected by this change, and is still off by default.)

- **Case 12614**

On LINUX the file selector has been replaced with a new one that allows multiple file filters to be set. This means that it is now possible for the file selector to show all *.PTF* and d3plot* files at the same time.

- **Case 20046**

Version 11 of D3PLOT supports a number of different 'interface force' files that LS-DYNA can now generate. Previously D3PLOT only supported and looked for the CTF (contact force files) associated with a set of PTF files but LS-DYNA can now generate Blast Force Files, FSI Force Files, CPM Force files and DEM Force Files that.

Although D3PLOT can read all of these file types only one of these can be read at any time along with the PTF files. The 'Open Model' menu now contains a popup menu that allows the user to select which of these additional files is read if they have run an analysis that generates more than one of these files.

- **Case 17982**

D3PLOT V11 can read an extended range of 'contact force' files. It handles all known LSTC formats of this file as written by LS971 R6, and searched for them in the order:

(1) jobname.ctf

(2) ctfile

(3) fsifor

(4) blstfor

Other files of the same format may be readable if renamed to one of the above.

Scripting

- **Case 20101**

D3PLOT JavaScript's can now use the 'Use' statement to split a script up into separate files. Additionally scripts can now use Unicode.

- **Case 18753**

See case 19674

- **Case 18423**

See case 19674

- **Case 19674**

Added a new function GetModelInfo() to the JavaScript API which returns an object with data fields:

.PTF_name full pathname of PTF (d3plot) file

.ctf_name ditto for CTF (intfor) file

.ZTF_name ditto for ZTF file

.xtf_name ditto for XTF file

Also:

.num_families number of file families in an adaptively remeshed analysis

.num_states number of states in a file sequence.

(See also cases 18753, 18423, 18171, 17979)

- **Case 18171**

See case 19674

- **Case 18434**

The functions GetCurrentDirectory() and GetStartInDirectory() have been added to D3PLOT JavaScript.

- **Case 17979**

See case 19674

Sections
- **Case 16218**

The shape of beam sections were not being drawn correctly if they had been defined on the *ELEMENT_BEAM_SECTION card rather than the *SECTION card.

This data is now written to the ZTF file and used by D3PLOT to draw the correct section shape.

User interface
- **Case 14333**

The results of MEASURE can now be dragged over for subsequent cut and paste, see case 13997.

The same 'angle with global' output as shown in PRIMER for node-to-node measure is now also shown, giving the angles between the vector between the nodes and the global XY, YZ and ZX planes.

User defined

- **Case 19427**

When User-defined data components are imported from externally generated ASCII files solid and shell tensor data are expected to be in the global coordinates system. This present a problem for users whose data originates from other FE codes, notably Nastran, which output shell results in the element local system since the conversion of a tensor from element local to global systems is not a trivial operation.

Therefore the following optional keywords have been added to the ASCII file user input:

GLOBAL This is the default and need not be specified, but is included for completeness.

LOCAL_DYNA This implies that the external data is in the local element system as understood by LS-DYNA.

LOCAL_NASTRAN_3D This implies that the external data is in the local element system as understood by NASTRAN, and all 6 tensor terms will be supplied.

LOCAL_NASTRAN_2D Is the same as LOCAL_NASTRAN_3D, but assumes that shell data will be provided and only the 2D (in-plane) tensor terms XX, YY and XY will be provided.

If these are defined the relevant 'local to global' transformation will be applied when the data are read so that the data stored inside D3PLOT are in the correct global system.

The reason for the separate DYNA and NASTRAN local systems is that the two codes have subtly different definitions of the element local system for shells and brick hexas, and extremely different definitions for degenerate wedge and tetra solids.

Views

- **Case 18249**

In D3PLOT the Caps Lock key can be used to tie dynamic rotation in all windows together, but this has the disadvantage that if accidentally left on it results in all text, in any application, being capitalised.

Therefore it is now possible to the Num Lock (top left of the numeric keypad) key to achieve the same effect. This has far fewer side-effect, and generally won't affect anything else. (The Caps Lock key still works in this context, either or both may be used.)

Write

- **Case 19784**

Dialogue commands have been added to write initial keyword data (WRITE->KEYWORD).

3.3   T/HIS

3.3.1   Enhancements in 11

Curve menu

- **Case 18205**

In version 11 the curve table option in T/HIS has been completely rewritten and now includes a lot of new options. The curve table now contains more information about each curve (curve ID, title, model, entity type, entity ID, component) and is presented in a similar way to the PRIMER Part Table. Each column of information can be turned on/off as required and the width and position of all the columns can be adjusted.

The data within the table can be sorted by any of the columns by clicking on the column header and the list of curves displayed in the table can be filtered by: Model ID, Entity Type, Component or Label.

In addition to adding/removing curves to and from graphs the curve table can now we used to

1) Select multiple curves and then add them to existing groups (or create a new group)

2) Change the colour, line style for curves

3) Delete curves.

Curves

- **Case 20863**

The 'Blank By Model', 'Unblank by Model' and 'Delete by Model' curve selection menus now contain an option to toggle between displaying model names and directories to make it easier to choose which curves you want to select.

Data access

- **Case 16799**

From version 11 onwards T/HIS has 2 new options to make it easier to re-plot curves from analysis that are still running.

Option 1

========

The 'Read LS-DYNA' menu now has a new option

'Overwrite existing curves'

which can be selected when the option to automatically extract results to match the curves from an existing model is used.

If for example the user has read in some curves from a model (and possibly carried out some curve operations on them) while the analysis is still running then he can use this option along with 'Re-Read' to re-open the model files, extract the same curves and then overwrite the original curves with the new ones.

The 'overwrite' option can also be used when opening a second model (instead of re-reading the 1st one) to replace the curves from the 1st model.

Option 2

======

Quick Pick now has a 'Update Curve' option. This option automatically re-reads any models for the curves selected and then regenerates the curves.

Edit

- **Case 19479**

From version 11.0 onwards the EDIT menu had been modified so that as well as being able to drag a single point a group of points can also be dragged. The group of points is selected by dragging out an area using the right mouse button. Once the area has been selected the points are highlighted on the screen and in the scrolling list. The user can then choose to drag the curve in X and Y, X only or Y only.

Once the point(s) have been selected they remain selected even after the mouse button is released so the same set of points can be moved multiple times without having to reselect them. To deselect the points so a new set can be selected the user can either

1) Press ESCAPE in the graphics window

2) use the middle mouse button

3) Use the new 'Unselect Points' button in the edit menu.

- **Cases 7979 & 16420**

From version 11.0 onwards the EDIT menu had been modified so that as well as being able to drag individual points the whole curve can also be dragged. As with selecting a single point the user can choose to drag the curve in X and Y, X only or Y only. As the curve is dragged the scrolling list showing the points updates to show the new coordinates of each point.

- **Case 6284**

From version 11 onwards the T/HIS curve EDIT menu has been modified. When deleting points they are no longer deleted as you click on them in the scrolling list instead they are now highlighted in red in the menu (and marked on the screen) and are not deleted until the user selects the new 'Delete Points' button. As well as selecting individual points a range of points can now be selected by clicking on the first one and then scrolling to the last point required and clicking on that.

Fasttcf

- **Case 13653**

By default a FAST-TCF script would terminate if there were more than 10 errors.

Version 11 has a new preference 'ftcf_error_count' that can be used to change the default value of 10 to anything from 1 to 999.

An option has also been added to the FAST-TCF 'run' menu which allows the number of errors to be changed interactively.

- **Case 14873**

Version 11 of T/HIS has a new "style_m" command that can be used to set the line styles for all of the curves belonging to a model. The format of the command is the same as the "stylec" command that sets the style for a single curve except that the 2nd argument is a model number instead of a curve tag.

style_m 3 red,default,solid,square,1

sets all the curves belonging to model 3 to

colour : red

line width : default

line style : solid

symbol type : square

symbol freq : 1 point

- **Case 11312**

From Version 11 onwards a dialogue command option for running a FAST-TCF script has been added.

The new command is

/FTCF RUN <filename>

i.e

/FTCF RUN e:\default.inp

Menus

- **Case 18990**

In version 11.0 the directory selector menu has been replaced with a new one that includes a top bar into which a path can be pasted.

- **Case 19016**

From version 11.0 onwards the Legend settings can also be accessed via both the DISPLAY menu and the DISPLAY dropdown menu.

- **Case 19017**

In version 11 the wording in the Legend Options menu has been changed to make it more obvious what the different Layout options do.

'Default' has been changed to 'Column List'

'Full' has been changed to 'Off' (as it turns the legend off)

- **Case 19021**

From version 11.0 onwards the Graph Tab in the top left hand corner of each Graph now has a popup containing the 'Only', 'Raise' and 'Delete' options that are in the main graph selection menu.

New features

- **Cases 14904, 14732, 11673**

From version 11.0 onwards DATUM lines can be defined and plotted in graphs. T/HIS will allow a maximum of 256 DATUM lines to be defined and each line can be added to any graph. The DATUM lines can be defined as either

      a) Constant Y value

      b) Constant X value

      c) A curve (maximum of 1000 points).

For each DATUM line the colour, line style and width can be defined along with a label and the labels position.

As well as defining a line the area above/below ort left/right of the line can be filled using any of the standard T/HIS colours.

If multiple DATUM lines are defined on a graph they are added to the graph in the order they are defined.

Operations

- **Cases 12975 and 18618**

From version 11.0 onwards the VEC operation in T/HIS has been modified to take 3 groups of curves as input. The 3 groups should contain the same number of curves and the curves are used in order of ID's.

If group 1 contains curves #6,#3 (picked in that order)

group 2 contains curves #4,#7 (picked in that order)

group 2 contains curves #8,#9 (picked in that order)

then 2 curves will be produced as output

VEC(#3,#4,#8)

VEC(#6,#7,#9)

The VEC(2d) option has also been modified to take 2 curve groups as input.

- **Case 17195**

From version 11 onwards the curve operation ZERO (which translates the curve so that the 1st point is at 0,0) now has options to either translate in X and Y (the default), X only or Y only.

Quick pick

- **Case 16802**

From version 11.0 onwards the following curve functions that take multiple curves as input can now be access from the Quick Pick menu.

> SUMmation
>
> ENVelope
>
> MINimum
>
> MAXimum
>
> AVErage
>
> RESultant

All of these functions produce a single output curve which is always written to the first free curve.

Read

- **Case 17785**

If the option to 'extract curves to match model' is used when opening a second or subsequent model then a new option has been added to create the curves using the default T/HIS line styles.

Previously the user had to select between either copying the curve styles from the curves he was matching or to specify a curve style for all the new curves. Both of these options could result in lots of curves with the same styles being created.

Using the new option results in each curve being given a different colour and line style.

- **Case 19663**

Version 11 supports the DCFAIL ASCII file and the DCFAIL data in the the LSDA (binout) file.

This file contains additional data for spotweld solids and clusters models using the _DAIMLERCHRYSLER version of *MAT_SPOTWELD (this version of the material does not support beam elements).

The DCFAIL file contains additional failure data showing how close to failure the spotweld is in tension, shear, bending and torsion, In addition it also contains another copy if the normal spotweld forces written to the SWFORC file.

The new data components appear under the SOLID and ASSEMBLY sub types within the SPOTWELD menu. If the SWFORC file is also present then the

normal forces and read from the SWFORC file, if the SWFORC file doesn't exist but the DCFAIL file does then the data components (Normal, shear forces etc) that are mirrored in the DCFAIL file are read from there instead.

As the DCFAIL file only contains the ID's and not the types or each spotweld then it is not possible to tell from the DCFAIL file alone which items are solids and which ones are spotweld clusters. If the SWFORC file is preset then T/HIS used the information in this file to match up the ID's and work out the type of each item in the DCFAIL file. If the SWFORC file isn't present then it attempts to use the data in the ZTF file to work out the types.

- **Case 18980**

Version 11.0 supports the ELOUT_PSD and NODOUT_PSD data files that are written to the binout (LSDA) file.

- **Case 18979**

Version 11.0 supports the ELOUT_SSD and NODOUT_SSD data files that are written to the binout (LSDA) file for analyses that use the *FREQUENCY_DOMAIN_SSD option in LS-DYNA

- **Case 19102**

T/HIS 11.0 supports the extra time history variables for elements that can be written to the LSDA (binout) file in later versions of LS-DYNA 971.

- **Case 20941**

Version 11 of T/HIS now supports the DISBOUT file written for Discrete Beams. Both the ASCII and the LSDA (binout) formats are supported.

The data components available in this file can be accessed via the BEAMS menu in T/HIS.

- **Case 18993**

T/HIS 11.0 supports the ELOUTDET file if it is written to the LSDA (binout) binary file. The ASCII version of ELOUTDET is not supported.

At present the option to output element stresses and strains at the elements nodal points instead of the elements integration points is not supported.

Element strains at surfaces other than Top, Middle and Bottom are also not supported.

- **Cases 17937 and 11835**

In version 11 two new options have been added to help filter the list of curve files returned by the 'Search Directories' option.

1. There is now an option to change the file extension used when searching directories. By default this is set to '*.cur' but this can be changed to any other files extension.

    '*.xy' would list all the files with the file extension '.xy'

    '*run.cur' would list all the files with a '.cur' extension which had a file that ended in 'run'.

basic_run.cur - would be listed

basic_test,cur - wouldn't be listed

2. A separate 'Filter' option that can be used to filter out unwanted curves. The filter works by matching against the full pathname of the file. If for example the following files were found by setting the file extension to '*.cur'

../run1/disp.cur

../run2/disp.cur

../test1/disp.cur

../test2/disp.cur

Then setting the filter to '*run*' would display only

../run1/disp.cur

../run2/disp.cur

and setting the filter to '*2*' would display only

../run2/disp.cur

../test2/disp.cur

- **Cases 18919, 18858 and 14752**

Version 11.0 of T/HIS can read data from the following DIAdem format data files.

REAL32	WORD8
REAL48	WORD16
REAL64	WORD32
INT16	ASCII
INT32	

The MSREAL32, TWOC12 and TWOC16 file formats are not supported.

If the DIAdem data file contains a time channel then that can be selected as the X-axis for all the other channels. If it doesn't then a start value along with an increment can be specified for generating the x-axis values.

- **Case 18265**

From version 11.0 onwards T/HIS can read the local X,Y and Z forces for nodal force groups from both the ASCII NODFOR file and the binout (LSDA) file. If a local force is requested for a node group that has not been defined using a local coordinate system then the global value is read.

- **Case 20942**

Version 11 of T/HIS now supports the PLLYOUT file written for Pulley Elements. Both the ASCII and the LSDA (binout) formats are supported.

The data components available in this file can be accessed via a new PULLEYS option on the read menu in T/HIS.

- **Case 19106**

The 'Stonewall' menu in T/HIS now contains an ERERGY data component that is populated using the individual stonewall energies that are written to the GLSTAT file. This data is a available from both the ASCII files and from the LSDA (binout) file.

Scripting

- **Case 18431**

A property has been added to the Model object which returns the file that was selected when loading the results.

- **Case 18435**

The functions GetCurrentDirectory() and GetStartInDirectory() have been added to T/HIS javascript.

T/his link

- **Cases 17804 and 11527**

When using the D3PLOT > T/HIS link items can be selected in T/HIS by using the 'Pick Visible' option and then selecting them in the D3PLOT window. From version 11.0 multiple items can now be selected in the D3PLOT window buy dragging out an area in the D3PLOT window.

- **Case 18627**

From version 11.0 onwards the D3PLOT > T/HIS link now has new options to help locate the items in the D3PLOT window that a curve in the T/HIS window belongs to. Previously the only 'locate' option was to draw a cross hair or blob on the item in the D3PLOT window but this could be difficult to see in complex models. From version 11 onwards there are 2 new options in the 'Locate' menu.

Sketch - sketches the item in wireframe.

Only - blanks everything except the item.

User interface

- **Case 16025**

In version 11.0 the right hand menu area can be made wider by dragging the green separator bar. BY making the right hand menus wider the Curve Manager can display more of the curve labels to make it easier to identify curves for operations like Delete.

As well as making the right hand menu wider the curve manager can also be undocked and it can also be dragged out of the main T/HIS window and placed on the desktop. When the curve manager window is undocked or on the desktop it's width can changed as required.

Write

- **Case 19213**

The WRITE menu has been modified so that if there is only one curve then that curve is automatically selected.

3.4   REPORTER

3.4.1   Enhancements in 11

Capture

- **Case 13034**

Multiple D3PLOT images can now be inserted into REPORTER from one D3PLOT object, eliminating the need to have multiple D3PLOT objects to produce multiple images from one analysis.

- **Case 8095**

PRIMER objects have been added to REPORTER.

- **Case 7779**

After a 'Capture' in REPORTER, the image produced is automatically inserted into the report, without having to 'Generate' the object.

Generate

- **Case 18488**

The order that items will be generated in on a page is now shown by a number on each item. This can be hidden if required.

Image

- **Case 17601**

Animated gifs now work in PowerPoint output.

Multiple jobs

- **Cases 7712, 8956, 9020 and 10742**

Multiple reports can now be combined into a single PowerPoint/html/pdf file.

Scripting

- **Case 15550**

It is now possible to specify a JavaScript to run in a D3PLOT object.

User interface

- **Case 18489**

The order that items will be generated in is now shown as a number at the top right of items if they can be generated.

Variables

- **Case 21417**

A new library script has been added to write a variable file

Write

- **Case 11858**

PowerPoint (pptx) files can now be written directly from Reporter.

3.5 Shell

3.5.1 Enhancements in 11

General

- **Case 21453**

A model list file can now be specified for D3PLOT in the D3PLOT->OPTIONS menu.

- **Case 20462**

The CPM and DEM options have been added to the Optional Files panel.

- **Case 16408**

The GMINP and GMOUT options have been added to the Optional Files panel.

User interface

- **Case 18052**

If you left clicked on the Reporter button it would start interactively as expected.

If you right clicked and opened the options menu, filled in some details for a job, the next time you left clicked on the Reporter button, the options menu opened rather than starting Reporter. This is not what people expected.

From v11, if you left click on the Reporter button it will now always start Reporter interactively, regardless of what is selected in the options menu.

Write

- **Case 18904**

New preferences have been added to turn on/off the Reporter output options:

shell*write_reporter
shell*write_pdf
shell*write_html
shell*write_vba
shell*write_ppt