

Solution Explorer in LS-PrePost - a GUI for Nonlinear Implicit FE

Thomas Borrvall¹

Wenhui Yu², Chengju Zhang², Zhanqun Zhang², Anders Jernberg¹

Anders Jonsson¹, Satish Pathy²

¹DYNAmore Nordic AB, Linköping, Sweden

²LSTC, Livermore, CA

The evolvement of multiphysics capabilities in LS-DYNA has made it a very powerful, albeit somewhat complicated, simulation product. The learning curve is steep, which makes tutorials, examples and classes essential to even get started. The Solution Explorer in LS-PrePost was introduced a few years ago to partly rectify this issue in the context of fluid mechanics, and this has now been complemented with a framework for setting up nonlinear implicit mechanics problems. The vision of the Solution Explorer is to combine simplicity and power in a strongly integrated pre- and post- environment for multiphysics, and this paper presents its current state for the special case of nonlinear implicit thermomechanics. We intend to cover the entire chain from model setup to post-processing for both single and multiple cases, in hope that it will provide a clear picture of its potential.

1 Introduction

The history of LS-DYNA [1,2] began at Lawrence Livermore National Laboratory (LLNL) in the 70's when Dr. John O. Hallquist made the ingenious decision to implement a 3D finite element code, called DYNA3D, to suite the purposes of the defense industry. This was indeed a bold move, and probably to the dismay of his superiors, considering that computer powers at the time were at most adequate for simplified simulation concepts and not quite ready for the CPU intensity required by a fully-fledged finite element software. With this in mind it is not hard to see that much of the development work had to be devoted to something that has become one of the LS-DYNA legacies; *speed*. Then and today, the design of algorithms is towards optimal complexity and best performance when executed on modern computer architectures. Over the next decade, the rapid evolvement of computer technology

brought DYNA3D into the spotlight at the eyes of the industry world and lead to the founding of Livermore Software Technology Corporation (LSTC) and birth of LS-DYNA at the end of the 80's. Since then LS-DYNA has been an invaluable tool in conceptual design, in particular a work-horse in the automotive industry, and is a major reason why simulations has earned its place in computer history [3].

The LS-DYNA origins lie in highly nonlinear, transient dynamic finite element analysis using explicit time integration, and this is the trademark it is still very much renowned for. Implicit technologies were introduced in the late 90's to cover a broader range of applications, and this also opened the door for addressing multi-physics on a larger scale. A vision of LSTC emerged, to provide the industry with a tool that allows engineers to work with a single model to conduct their simulation projects, without being dependent on other software. While this idea is conceptually simple and straightforward, its practical implementation is not and success in the end relies on a number of factors. The strengths of LS-DYNA in this aspect include in particular its ability to simulate complex real-life problems where several physical phenomena interact, for instance the coupling between structural deformations, thermal conduction, fluid flow and electromagnetic effects. Needless to say, this complexity demands interdisciplinary knowledge to set up a functioning model, e.g., selecting solution parameters, material properties and boundary conditions for the involved physical disciplines. On the one hand, the LS-DYNA Keyword User's Manual [1] constitutes a complete coverage of all fields, but serves more as a dictionary for experienced LS-DYNA users rather than a source for learning how to use the software at a beginner or intermediate level.

Without degrading the importance of introductory classes and online tutorials, which will always have a deserved place, we feel there is need for a complementary product that lowers the threshold for entering the world of LS-DYNA. The Solution Explorer in LS-PrePost, a high level GUI for model setup and post-processing, was introduced a few years ago as a mean to fill this gap in the context of fluid mechanics. During the last two years work has been devoted to add implicit thermo-mechanics to this interface, and this paper focuses on this latter part. We present the general idea and overview of its main components in Section 2 which is followed by a tutorial in Section 3. The paper ends with some conclusions and visionary thoughts in Section 4. We emphasize that presented figures and component details are destined to be outdated, but still hope that the philosophy and basic concepts will be made clear and remain timeless. The purpose is primarily to provide a taste of what the future might bring.

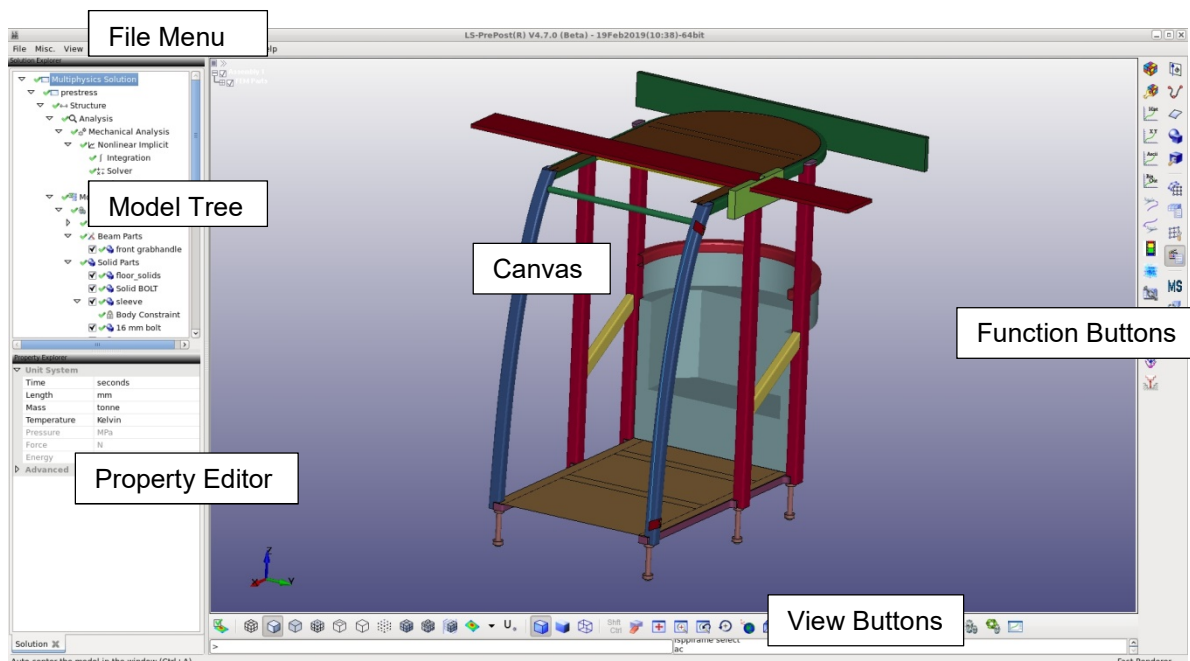


Figure 1 Snap shot of LS-PrePost and Solution Explorer

2 The Solution Explorer

There are many powerful pre-processors that support LS-DYNA, and LS-PrePost can be used for post-processing, so it is justified to ask what the Solution Explorer adds to the mix. Before going in to specific features, we refer to Figure 1 for an overview.

2.1 Overview

The Solution Explorer consists of the Model Tree and the Property Editor (*L-menu*), while the standard LS-PrePost functionalities are accessed through the Function Buttons (*R-menu*). The Canvas is shared by the two menus and displays the model while the View Buttons are primarily for view manipulations. The Model Tree is a representation of the model as a hierarchical tree structure, where the branches correspond to its components and/or features. Branches are in general equipped with properties that are editable in the Property Editor whose appearance adapts to the specific feature selected in the tree. The purpose with this paper is not to go through each branch in detail, but to get a feel for the layout we here present the top levels of the tree.

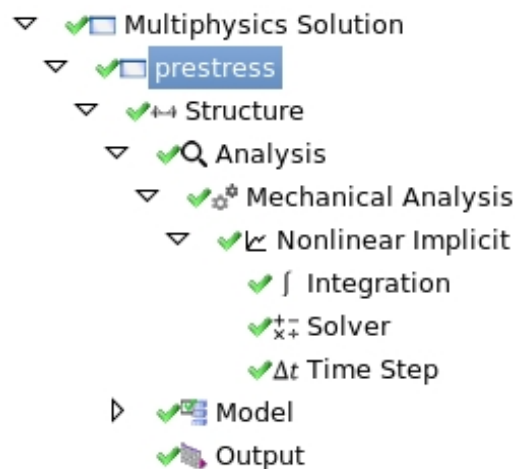


Figure 2 Model Tree example, with a single case (prestress) using nonlinear implicit structural analysis

Level 0	Multiphysics Solution	Top level, select unit system for the solution as a whole and right-click to save and/or run.
Level 1	Cases	A solution may have several cases that are executed in sequence. Set overarching properties of each case such as termination time, and optionally run from a given case.
Level 2	Structure/Fluid	A case can incorporate a structure, a fluid or potentially both, determining the solution type for this case; structure, fluid or fluid-structure interaction.
Level 3	Analysis	Set analysis options for the structure and/or fluid, mechanical and thermal options can be selected and specified.

<i>Model</i>	<i>Contains the model with all its parts, contacts, connectors and boundary conditions, the core of the solution.</i>
<i>Output</i>	<i>Specify general output options that are not associated with a specific branch elsewhere.</i>

Treatment of levels 4 and below is limited to what is encountered in the tutorial presented in Section 3. We now devote the rest of this section to attributes that makes the Solution Explorer stand out in comparison to other pre- and post-processors.

2.2 Features

2.2.1 Keywords

First, a common theme among pre-processors, including LS-PrePost itself, is that they are oriented around the *keyword* language. Keywords are the smallest constituents that make up an LS-DYNA input deck, and the combination thereof is used to represent concepts at a higher level of abstraction. This is termed a *bottom-up* approach, as small building blocks are at the core of the modeling philosophy. While this is reasonable if you are an LS-DYNA expert or at least have had some experience with it, for newcomers it is not as attractive. The Solution Explorer is aimed at circumventing this obstacle by making use of a terminology common to simulation engineers in general, thus suppressing the need for keyword linguistic skills. The idea with this *top-down* approach is that basic knowledge of continuum mechanics and finite elements, earned from university education or software experience, should be enough to get started with LS-DYNA.

2.2.2 Defaults

Continuing, the amount of keyword parameters to set or adjust in LS-DYNA is abundant. Fortunately only a fraction of them requires attention for conducting simulations within a specific application field, since most do not even apply and the remaining ones are presumably equipped with decent default values. However, the selection of these defaults is for historical reasons based on increasing execution speed rather than increasing accuracy, and often users benefit from alternate setups whose optimality is not easily obtained. This is particularly true in a nonlinear implicit context, for which specific and elaborate guidelines can be found online [4] or in the manual [5]. Unfortunately not many are aware of this material, or are for some reason not susceptible to its content, which leads to unnecessary burden for support teams among the LS-DYNA distributors. The Solution Explorer is designed to follow the mentioned guidelines to the greatest extent possible, and by that increasing the probability to achieve successful simulation results.

2.2.3 Simplicity

Not only can it be difficult to tune global control parameters in LS-DYNA, but for seemingly intuitive features LS-DYNA sometimes demands a great deal from users. For instance, specifying contacts offer many possible routes to take and sometimes it is hard to see the forest for the trees. In Solution Explorer a contact is defined by a minimum amount of parameters that still (presumably) covers most situations of interest. This design choice is based on the idea that a contact definition after all is conceptually simple and should not require excessive tuning for each individual application. Another example is pre-stress of bolts, which in LS-DYNA requires a combination of keywords for a complete definition. In Solution Explorer it principally suffices to specify the parts and the preload magnitude and details pertaining to the LS-DYNA keywords are transparent to the user. How the interfaces to various objects and features of this kind develop in the future will depend how well they are received among potential users.

2.2.4 Integration

Furthermore, pre- and post-processors are usually separate software focused on disjoint phases of the simulation process. The workflow typically involves (i) setting up the model in a pre-processor, (ii) solving the problem using the solver and (iii) viewing results in a post-processor. As far as (iii) goes, the results from LS-DYNA are scattered throughout a number of binary database files, ranging from

d3plots (animation) to binouts (plotting), and for newbies it is not easy to know where to fetch which quantity. Furthermore, it is not clear how result data in its rather primitive format should be organized and linked to high level objects in the pre-processing phase (i). In the Solution Explorer, the pre- and post-processing phase is strongly integrated and there should be little or no ambiguity as for the interpretation of results or where to find a particular one of interest. Branches are namely, when applicable, partitioned into a pre- and a post-section with simple and intuitive exposition of their respective content. The (i) and (iii) of the workflow presented above are therefore consolidated to a single session, without having to open any result file, and (ii) is connected via a temporary departure to LS-Run for executing or submitting the job.



2.2.5 Cases

A complex simulation process is favorably split into simpler tasks carried out in sequence. The purpose is to introduce an organization that simplifies the overall model setup. An example of such a setup can be seen in metal forming, where 'tasks' include placing, wrapping, forming, trimming and quenching the blank. In LS-DYNA there are a number of ways to implement this strategy, and the most popular one is using the *dynain* approach. Here a task is associated with a single LS-DYNA run and the state of the system is transferred between executions through file i/o. The state file (called *dynain* in everyday speech) is in keyword format which allows for an unprecedented flexibility when compared to other approaches, but setting up the simulations based on keywords alone can still be cumbersome. The Solution Explorer supports this approach by allowing multiple cases, with each 'case' being synonymous to a 'task'. A case appears as a sub-tree in the Solution Explorer and can be set up more or less independently of any other case, and the linking of two subsequent cases is handled automatically.

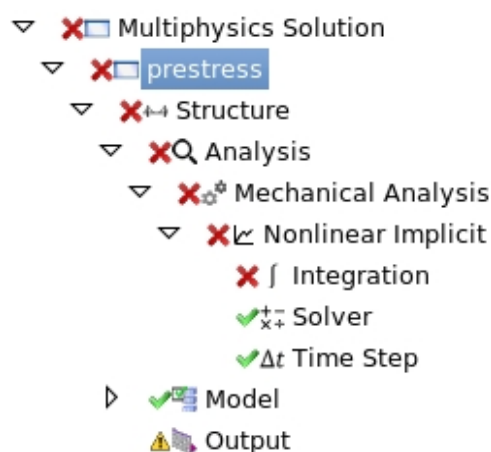


Figure 3 Model Tree example with error and warning

2.2.6 Errors and warnings

Finally, the Model Tree is equipped with an error and warning check to inform the user if the model is not runnable or some parameter selection is questionable. Figure 3 shows how it may look in the Model Tree, a red cross on a branch indicates an error and this cross propagates all the way up to the top level so as to not hide the error if menus were collapsed. In this particular example something is wrong in the *Integration* branch of the *Nonlinear Implicit Solver*. Like-wise a warning triangle means that some setting is not recommended or maybe redundant, and in the above example some value in *Output* needs to be reviewed. More details about the warning or error is found by opening (clicking on) the low level branch in the Property Editor. The trigger sensitivity of warnings is to an extent influenced by the opinion of the software designer, and may therefore in the future be optionally switched off. The obvious goal with this capability is for users to assess the state of the model on the fly without consulting special tools for the purpose.

2.3 Disclaimers

2.3.1 Mesh

The Solution Explorer is not a meshing tool, it is rather assumed that the mesh has been set up *a priori* and is ready to be operated on. Operations include adding various kinds of mesh connections,

e.g., joints and nodal rigid bodies, which as a side effect may result in creation of new nodes. This is however the sole mesh modification there is within the Solution Explorer framework. In LS-PrePost there are mesh functionalities among the Function Buttons (*R-menu*) and the aim is to integrate this with Solution Explorer in the sense that new mesh components are automatically imported into the Model Tree when focus is switched back to the Model Tree or Property Editor (*L-menu*).

2.3.2 Materials

The Solution Explorer interfaces to a material database, graphically represented as a tree organized with respect to physical characteristics. The current way it appears to the user is as shown in Figure 4, and the purpose is of course to equip parts with material models. Each physical material has one or more *tags* to say what characteristics should be modeled, including the specification of whether rigid, elastic or some other specific structural behavior should be assumed. If a thermal or coupled thermo-mechanical simulation is selected, the materials will also include the thermal properties. This material database is currently fixed and there is no opportunity to change material parameters while working on a model, the user simply selects a predefined material among the ones available. Material model selection and parameter fit is such a comprehensive field that deserves a software all of its own and the goal is here to not complicate the model setup with this additional task. However, the database is merely a text file called *material.xml*, formatted in combined xml and keyword and can be easily modified to incorporate any material model of interest in a project or in general. To this end, a default version of this database file is located in the LS-PrePost installation directory and when Solution Explorer is started this file is copied to the directory of the LS-PrePost configuration file. Once this copy exists, it will have precedence over the original one and consequently this is the file that should be edited for personal use. Admittedly, editing text files is a primitive task and the plan for the future is to load materials directly from the GUI and maybe even link to software for parameter fitting.

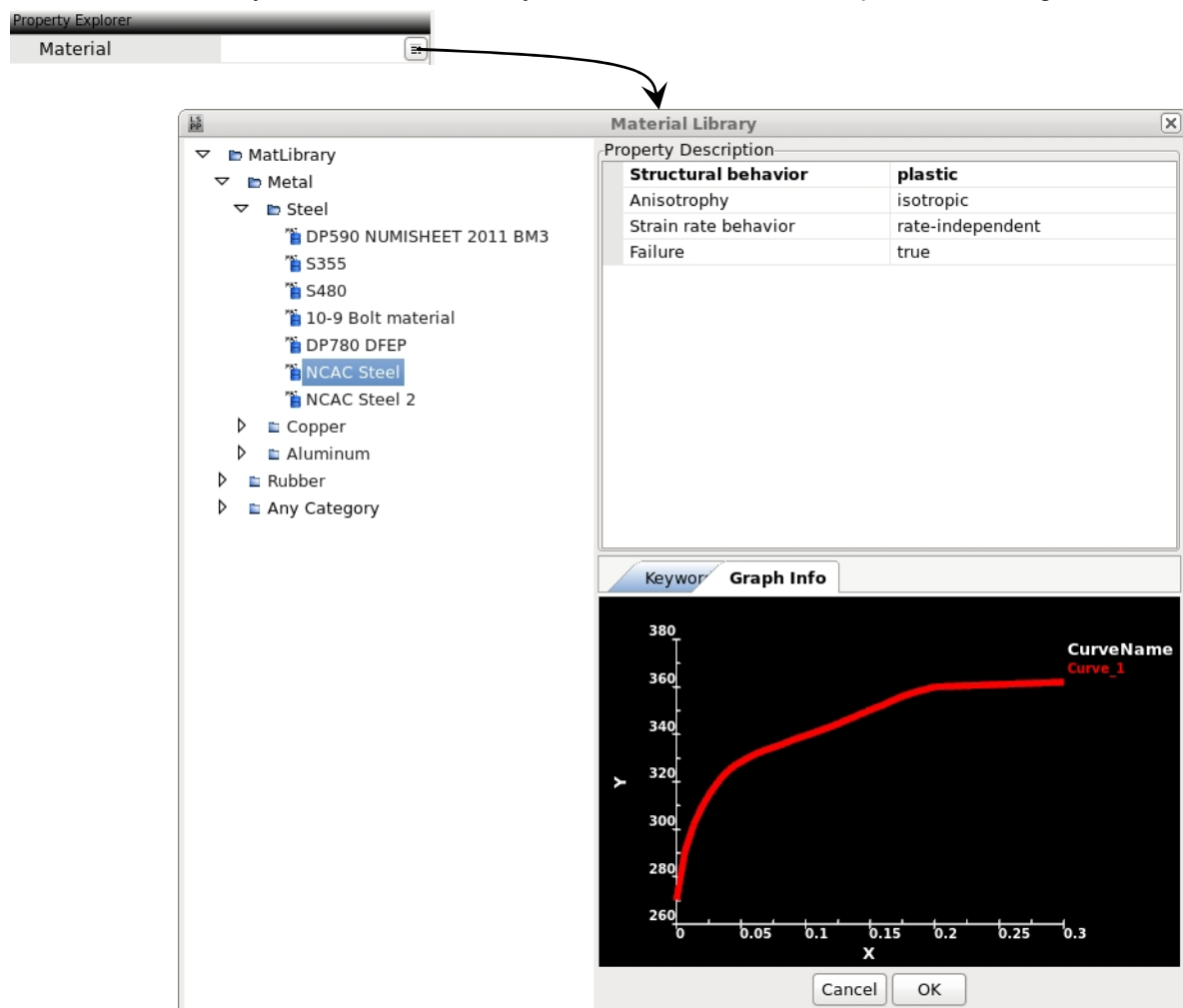


Figure 4 Customized material database for selection of material models

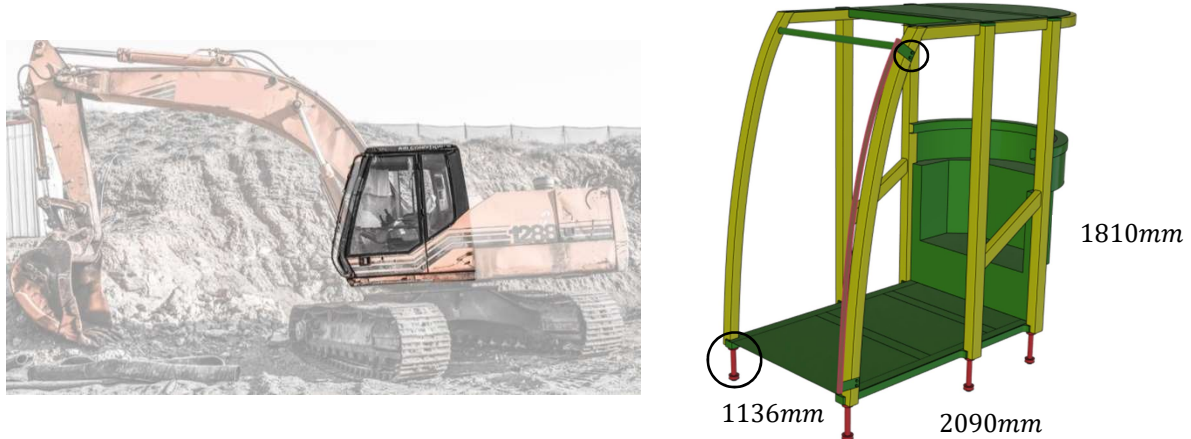


Figure 5 Cab for the protection of construction machine operator, bolts for prestress highlighted

3 Tutorial

For this tutorial you need the mesh input file *rops_mesh.k*, the customized material database file *materials.xml* and we suggest late versions of LS-PrePost, LS-Run and LS-DYNA [7].

3.1 ROPS

The tutorial example is inspired by the ISO requirement 3471 for Roll Over Protective Systems (ROPS) of earth moving machinery [8]. The analysis is performed on the model of a hypothetical cab design mounted on a ~15 tonne wheeled excavator. The geometry is illustrated in Figure 5 and the test is divided into a sequence of 4 load cases

1. The *prestress* case, 600 MPa pre-load of 6 floor and 4 grab handle mount bolts
2. The *side* case, lateral loading 200mm
3. The *roof* case, vertical loading 50mm
4. The *behind* case, longitudinal loading 150mm

of which cases 2-4 are shown in Figure 6. The *prestress* case is included for educational purposes and not part of the legal requirements, the two sets of bolts to be pre-loaded are highlighted to the right in Figure 5.

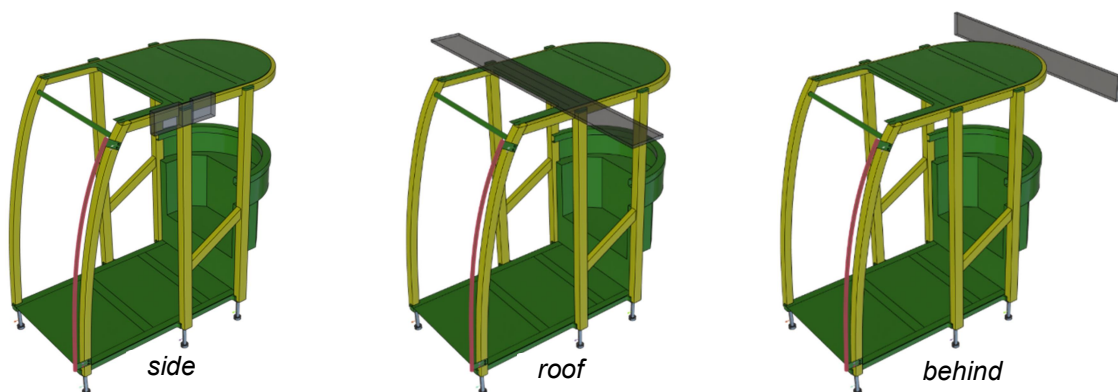


Figure 6 Rigid impactors with prescribed motion are used to conduct cases 2-4

3.2 Starting Solution Explorer

Solution Explorer is a software within LS-PrePost, so first you need to start LS-PrePost. The Solution Explorer is then found in the File Menu (see Figure 1) through *View->Solution Explorer*, which will

open the Model Tree and Property Editor on the left hand side. If Solution Explorer becomes a part of your every day work, this is saved in the configuration file and will be the default view when starting LS-PrePost.

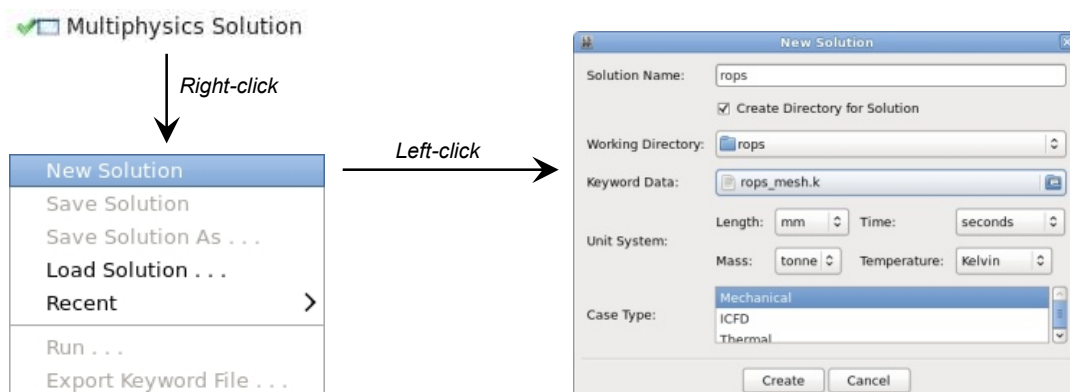


Figure 7 Starting a new solution from Multiphysics Solution branch

3.3 Creating a Solution

Even if you opened *rops_mesh.k* in connection with starting LS-PrePost (or through *File->Open->LS-DYNA Keyword File->rops_mesh.k* in the File Menu), this is not known to the Solution Explorer until you explicitly start a new solution to load the content of the file into the Model Tree. To this end, right-click on *Multiphysics Solution* in the Model Tree and select *New Solution* and this will bring up the pop-up shown to the right in Figure 7. Here you can select the *Solution Name* which will be the name of the solution database file (it will get the extension *.solution* by default) and also a new directory in which all data for this solution is stored. The *Working Directory* is essentially an existing path to where the solution directory is to be created. For *Keyword Data* you can either leave as (*None*) if you already opened the mesh file since the default is to import an existing model, or you can point to the mesh file on your file system. The *Unit System* should be *mm*, *seconds*, *tonne* and *Kelvin*. At this point it is only important to select the length unit correct, which should be the unit used when creating the mesh. The other units can be changed later in the Model Tree, while a change in length units will scale the mesh. The *Case Type* is here *Mechanical* since we are not interested in any coupling, but this is also something that can be changed in the Model Tree by adding and/or removing disciplines. Click *Create* and a *Case_1* Model Tree should be created. In the tree you can left-click on the label *Case_1* to rename it to *prestress* to reflect on what is to be done (pre-load of bolts), and then right-click on *Multiphysics Solution* and select *Save Solution* to save the solution file. Now the tree should look similar to what is in Figure 8. At any point you can revert to the latest saved Model Tree by right-clicking on *Multiphysics Solution* and select *Load Solution*, the tree will be replaced with the content of the solution file you decide to load. The *Run...* and *Export Keyword File...* options will be discussed in Section 3.5.1 and onwards. Now we are up and running and ready to start modeling the *prestress* case, but first some general remarks.

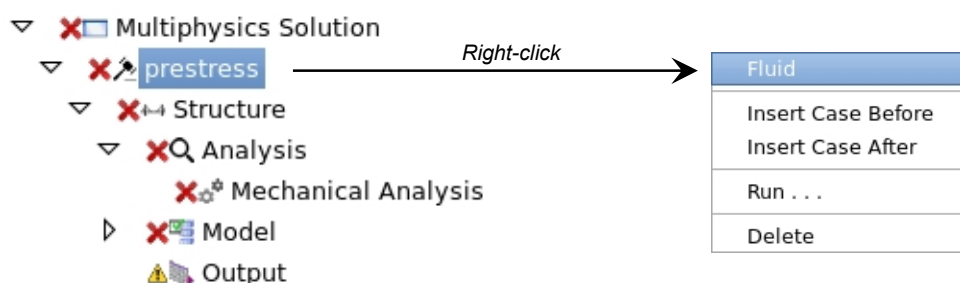


Figure 8 Initial solution tree

3.4 Units and Advanced Curtains

A recommended work flow in Solution Explorer is top-down, so it makes sense to start at the top of the tree. On the *Multiphysics Solution* branch, the current unit system is displayed. The unit system can be changed, which will transform all parameters in the tree accordingly. Therefore it is worth emphasizing that all parameters *must* be edited with respect to the unit system specified here. As a reminder while working, all parameters in the tree are conveniently displayed with units when applicable. Furthermore, you may also specify which LS-DYNA version the model is intended for. This option is hidden behind a closed *Advanced* curtain which needs to be opened to reveal the selected version. As the name suggests, *Advanced* curtains are intended for supplementary studies and are *always* closed if the hidden selections are the defaults. Vice-versa, if the default selections have been altered at some point the *Advanced* curtain will be *open* to reveal this deliberate change. This convention is followed throughout the Solution Explorer to clarify the appearance of menus in the Property Editor. For this problem we don't need to change unit system or output version.

Unit System	
Time	seconds
Length	mm
Mass	tonne
Temperature	Kelvin
Pressure	MPa
Force	N
Energy	mJ
Advanced	

3.5 Case Control Parameters

Proceeding to the case branch (we whenceforth call it *prestress* even if you happened to not follow this name suggestion), you need to specify the *End Time* and *Database Plot Interval*. No need for further explanations, select an end time of 1 seconds and a plot interval of 0.1 seconds, as this seems a reasonable time frame for preloading.

End Time	1 s
Database Plot Interval	0.1 s

3.5.1 Miscellaneous options

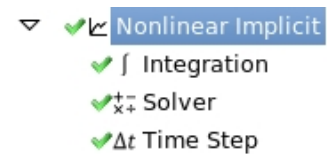
This is also the place for adding or removing any of the two supported materials, *Structure* and *Fluid*, see Figure 8. Right-click on *prestress* and you have the option *Fluid* which will add a parallel branch to the already existing *Structure*. The *Fluid* branch is for the fluid mechanics setup in case of ICFD or FSI analysis and not covered in this paper. Further options are to *Insert Case Before* or *Insert Case After* the *prestress*. This will clone the entire *prestress* tree to a *new* case tree on the same level, the new case will appear either before or after *prestress* depending on the menu choice. Once created, the new case tree can be renamed and modified appropriately. Selecting *Run...* in a case branch will run all cases in LS-DYNA *starting from the case you select it for*. More specifically, Solution Explorer will export the Model Tree to an LS-DYNA keyword input file, open LS-Run with a setup that allows you to execute this case and all subsequent cases with the system state being carried over between each case. Note that you also have the options *Run...* and *Export Keyword File...* on the *Multiphysics Solution* branch, see Figure 7, which by analogy will perform the corresponding operations on the *entire* model and not *only* for a selection of cases. On the case branch, as well as on the *Structure* or *Fluid* branches one level below, a right-click also gives you the option to *Delete* the corresponding branch.

3.6 Analysis

Proceeding down the tree, past *Structure*, we get to the *Analysis* branch. There are two types of analyses that can be performed on a structure, a *Mechanical Analysis* and a *Thermal Analysis*. Due to the selections made when creating the initial solution tree, see Section 3.3 and Figure 7, only the *Mechanical Analysis* is currently available as shown in Figure 8. However, a simple right-click on *Analysis* allows you to add the *Thermal Analysis* to the case and the entire case tree will adapt accordingly. This would for instance mean that thermal boundary conditions become available and thermal properties are required in selected features such as contacts, and so on. When both *Mechanical Analysis* and *Thermal Analysis* are present in the case tree, a coupled structural-thermal analysis is implicitly understood. Deleting an analysis type, achieved by right-click and *Delete* on either the *Mechanical Analysis* or *Thermal Analysis* branch, will remove all associated content in the case tree. For instance, if a flux boundary condition was created, it will be automatically deleted in connection to deleting the *Thermal Analysis* branch. A warning will be issued before this action is taken.

3.7 Nonlinear Implicit

Right-click on *Mechanical Analysis* to select *Nonlinear Implicit* to perform a nonlinear implicit analysis for the mechanical part of the problem. This is currently the only choice, in future we expect *Linear Implicit* and *Explicit* to be included among the available options. A *Nonlinear Implicit* tree will appear one level below, containing the branches



<i>Integration</i>	Time integration schemes, statics or dynamics
<i>Solver</i>	Nonlinear solver options, Newton parameters and tolerances
<i>Time Step</i>	Time step options, initial value and change over time

It should be noted that the *Thermal Analysis* type will contain the same three branches, since these are common for transient analyses in general. The actual content for each branch will vary slightly between *Thermal Analysis* and *Mechanical Analysis*, reflecting somewhat the physics and the availability in each respective analysis type. We here discuss the particular choices in the context of our tutorial and leave the rest for self studies.

3.7.1 Integration

We intend to constrain the structure to the extent that a static analysis should in principle be possible. However, before the bolts are pre-loaded there will be gaps in the contacts and risk of encountering singularities in the linear solver. Therefore we need to start with dynamics and switch to statics when the pre-load has had the effect of increasing contact stress. To this end, select *Types* as *Dynamics followed by Statics*, which in turn will allow you to set *Dynamics Ends* to 0.25 seconds and *Static Begins* to 0.75 seconds. Furthermore, when the intention is to obtain a (quasi-) static response it is always recommended to use numerical damping, so set *Methods* to *Newmark Damped* for a decent setting. This usually renders a smooth and stable solution behavior.

<i>Types</i>	Dynamics followed by
<i>Methods</i>	Newmark Damped
<i>Gamma</i>	0.6
<i>Beta</i>	0.38
<i>Dynamics Ends</i>	0.25 s
<i>Statics Begins</i>	0.75 s

3.7.2 Solver

The pre-selected *Solution Method* is a *BFGS Slightly Nonlinear* method, and as the name suggests it would be a decent choice for problems with not much nonlinearity involved. For the pre-loading we expect stress levels in materials to be in the elastic region, and even though contacts play a role we deem them unchallenging because of small relative motion between parts. We therefore leave this selection as is. The *Tolerance Type* is set to *Recommended*, which means that the relative displacement change should be less than 0.001 for convergence. Incitement to changing tolerances is usually based on some knowledge of the problem, most often earned from investigation of log files from previous attempts. Here we don't have such information and therefore don't see a reason to change this setting.

<i>Solution Method</i>	BFGS Slightly Nonlinear
<i>Reformations Before</i>	15
<i>BFGS Iterations Before</i>	11
<i>Tolerance Type</i>	Recommended
<i>Relative Displacement</i>	0.001

Advanced

3.7.3 Time Step

While most branches are in error (indicated by a red cross), the *Analysis* branch becomes valid upon adding *Nonlinear Implicit* (indicated by a green tick). This is because this branch was added *after* the termination time was set to 1 second, see Section 3.5, and the initial time step is internally selected assuming the simulation is performed in 100 implicit steps. This results in an initial time step value of 0.01 seconds and is found as *Initial Size*. The initial time step may have to be adjusted manually depending on the application, and for the pre-load step we do change it to 0.1 seconds. If the *Nonlinear Implicit* branch had been added with the termination time *not* set, the *Analysis* branch would be in error since an initial time step estimate would not be possible and this is a mandatory parameter. Finally, to avoid time step reductions we set *Change Strategy* to *Aggressive*,

<i>Initial Size</i>	0.1 s
<i>Step Size Option</i>	Maximum Time Step Size
<i>Maximum Time Step</i>	0.1 s
<i>Change Strategy</i>	Aggressive
<i>Increase T.S. if iter <</i>	100
<i>Decrease T.S. if iter ></i>	110

which is to say that the time step will only decrease if the number of iterations to converge is greater than 110, and it will increase as soon as convergence is obtained in less than 100 iterations.

3.8 Model

The *Model* branch reveals the model content, and is made up of

<i>Parts</i>	All parts, including initial/boundary conditions imposed on them
<i>Initial Conditions</i>	Initial motion/temperature definitions on nodes
<i>Prestress</i>	Pre-loading definitions
<i>Boundary Conditions</i>	Prescribed motion/force/temperature definitions on nodes/segments
<i>Contacts</i>	Sliding/tied/transducer contact definitions
<i>Connectors</i>	Joint/rbe2/rbe3 definitions

where each component is represented by a branch one level down, see Figure 9. In the following we explain each category in the context of our tutorial, and this will hopefully inspire further investigations.

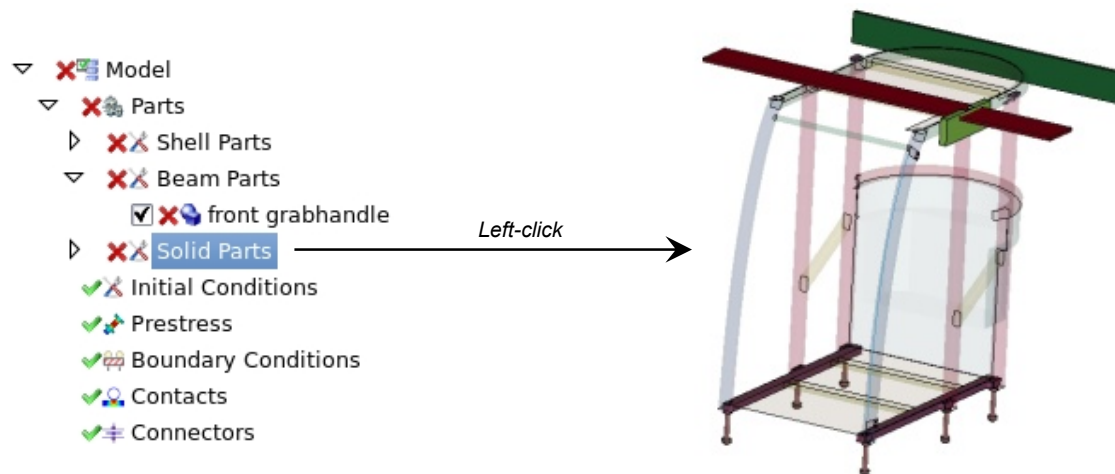


Figure 9 Model tree, with example of displaying solid parts while hiding others

3.8.1 Parts

Parts are divided into *Shell Parts*, *Beam Parts* and *Solid Parts* and each respective tree lists all parts of its kind in the model. Here we need to assign materials, thicknesses and boundary conditions, and also decide what parts to inactivate (if any) for the pre-loading. First let's discuss different ways to display the model in the Canvas since this has an impact on the way we proceed.

Part display

The highlighted branch or branches in the tree will determine what is shown in the Canvas. To this end, if *Model* or *Parts* is selected, the entire model is shown in shaded mode. From logical argumentation, if *Shell Parts* is highlighted then all the shell parts are shown, and the corresponding behavior holds for *Beam Parts* and *Solid Parts*. The parts that are *not* shown are not blanked out, but rather shown with transparency, which gives a good overview of the entire model while the selected parts are brought to the forefront, see Figure 9 for an example. Similarly, you may go down one level to select an individual part or parts to show only those while the others are sent to the background. Multiple selection in the tree is done by the standardized *CTRL+left-click* or *SHIFT+left-click* operations. Selection of parts can also be made directly in the Canvas, a left-click on a part will display it while a right-click will hide it. This latter way of doing things will work whenever the Canvas is not in some sort of picking mode, for instance when selecting nodes for boundary conditions.

Part assignments

As perhaps noted, the Property Editor has a common appearance for each respective part type. For all parts you need to select a *Material*, for shell parts you must also set a *Thickness* while for beam parts you define the cross sectional properties through a sequence of selections. In *rops_mesh.k* the element geometries are already defined so there is nothing to do for shell thicknesses or beam cross sections, but we need materials. The materials to use for each part is

<i>Material</i>	<i>Parts</i>	<i>Structural behavior</i>
<i>S480</i>	<i>grabhandle_tab</i>	<i>plastic</i>
	<i>top_horseshoe</i>	<i>plastic</i>
	<i>pillar</i>	<i>plastic</i>
	<i>front_pillars</i>	<i>plastic</i>
	<i>front_tube</i>	<i>plastic</i>
	<i>crossbeams</i>	<i>plastic</i>
	<i>mid_horseshoe</i>	<i>plastic</i>
<i>S355</i>	<i>U_beam</i>	<i>plastic</i>
	<i>roof_plate</i>	<i>plastic</i>
	<i>weld</i>	<i>plastic</i>
	<i>U-sheet</i>	<i>plastic</i>
	<i>U_beam floor</i>	<i>plastic</i>
	<i>floor_plate</i>	<i>plastic</i>
	<i>floor_solids</i>	<i>plastic</i>
	<i>pillar solid cover</i>	<i>plastic</i>
<i>NCAC Steel</i>	<i>front grabhandle</i>	<i>elastic</i>
	<i>sleeve</i>	<i>rigid</i>
	<i>ROPS Long</i>	<i>rigid</i>
	<i>LDD Lateral</i>	<i>rigid</i>
	<i>ROPS vertical</i>	<i>rigid</i>
<i>10-9 Bolt material</i>	<i>Solid BOLT</i>	<i>plastic</i>
	<i>16 mm bolt</i>	<i>plastic</i>

of which all are defined in the *material.xml* accompanying this tutorial. To complete the part properties, select all parts that are supposed to use the *S480* material and open the material database from the Property Editor. The material database is opened by clicking the icon on *Material*, see top of Figure 4. In there, simply select *MatLibrary->Metal->Steel->S480* in the left hand tree, and *Structural behavior* as *plastic* on the right hand side. Leave the remaining selections untouched and click *OK*. Now all the selected parts are equipped with the *S480* material using an elastic-plastic formulation. Proceed accordingly for the other materials, but mind that *sleeve*, *ROPS Long*, *LDD Lateral* and *ROPS vertical* are *rigid* and *front grabhandle* is *elastic*.

Part activation

Each part in the Model Tree has a checkbox which is checked by default, see Figure 9 where *front grabhandle* is exposed as the only beam part. This checkbox indicates whether this part is *active* (box is checked) or *inactive* (box is unchecked) for this case. Inactive parts will be unaccounted for when performing the simulation, which is convenient if they only constitute extra baggage. For our pre-loading we have no use of the three impactors that are intended for the external loading of the structure, so uncheck the boxes for *ROPS Long*, *LDD Lateral* and *ROPS vertical*. This operation will grey out these parts in the tree, see Figure 10, and also remove from the Canvas, as a confirmation that they will be ignored in the simulation of this case. This completes the model in terms of establishing part properties, we are now left with imposing boundary conditions and interactions.

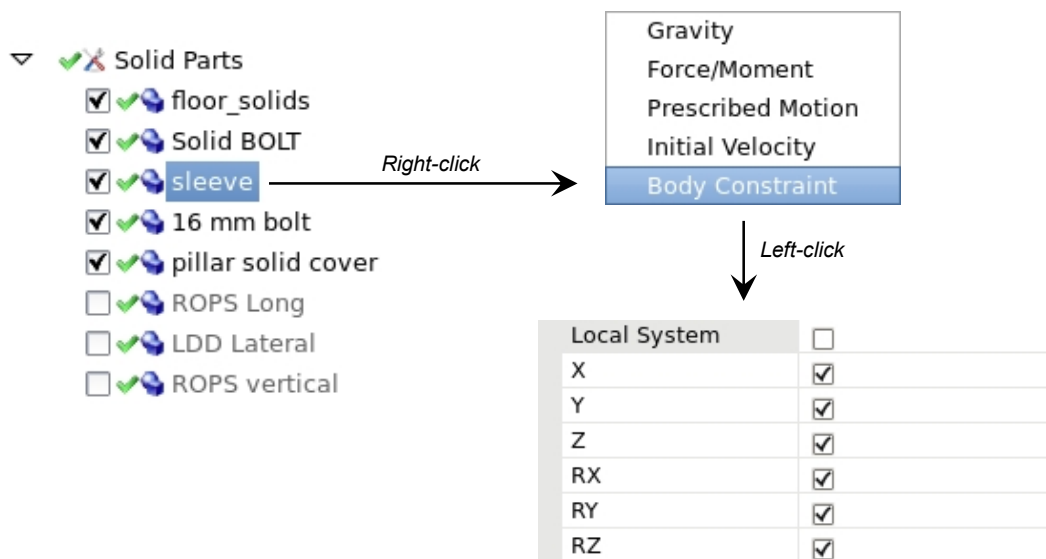


Figure 10 Inactivated parts greyed out in the tree, and fixing the sleeve in all directions

Part boundary conditions

On *rigid* parts only it is possible to assign initial and boundary conditions, such as prescribed motion and constraints. The *sleeve* in our model represents the attachment point on the excavator which we intend to be rigid and *fixed*. To this end, right-click on *sleeve* and you will see the available options, of which one is *Body Constraint*. Select it, and a *Body Constraint* will appear in the tree in association to *sleeve*. Select it in the tree by left-clicking, and the Property Editor will allow you to select how this body should be constrained. Since we want it pinned down we leave all degrees of freedom checked as shown in Figure 10. This is the only constraint needed for pre-loading case, and we will cover other types of conditions (prescribed motion) when dealing with the other cases in Section 3.10.

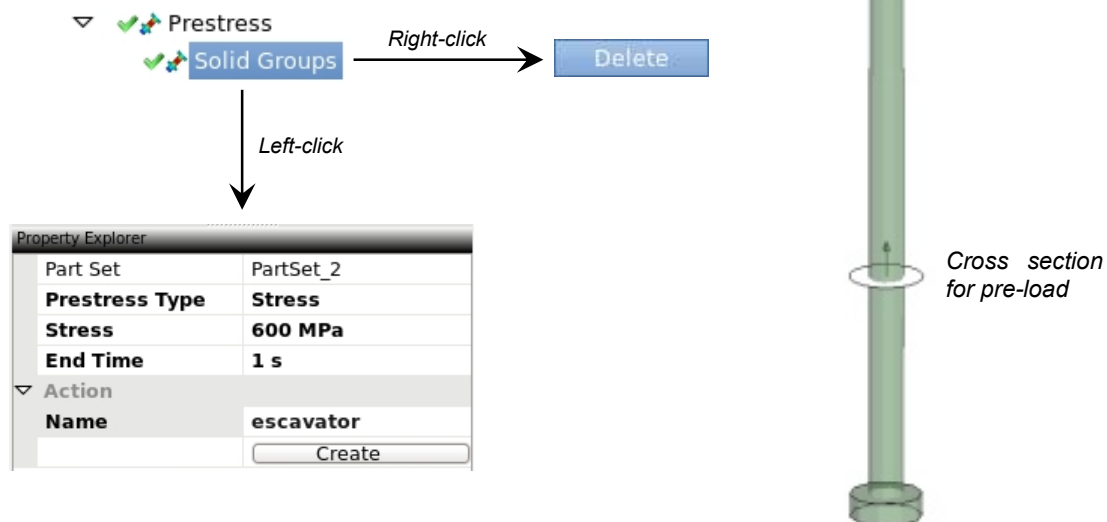


Figure 11 Creating pre-load for the 16 mm bolts

3.8.2 Prestress

Continuing, we need to pre-load the bolts using the *Prestress* branch. Right-click on *Prestress* and select *Solid Groups*, which will open a *batch dialogue* for defining arbitrarily many such *Solid Groups* (noting that we are yet to actually create an instance of such a group), see Figure 11. Corresponding batch dialogues are available for all features of whom there may exist more than one object, *Contacts* is another example that we will deal with in Section 3.8.3. A *Solid Group* consists of a *Part Set*, a

Prestress Type (the option of pre-loading with prescribed force or stress), the *Force* or *Stress* level and the *End Time* (the duration of the pre-load). For the pre-load of bolts by the *sleeve* (the attachment to the excavator) the *Part Set* of interest is the *16 mm bolt* which may be selected in the Canvas when *Part Set* is highlighted in the Property Editor. The *General Selection* in LS-PrePost is used to establish a part set in this context, and this is the way any such set is selected elsewhere too (node sets for boundary conditions for instance). The selection is completed by either pressing *Enter* on the keyboard, or clicking the green tick in the Property Editor. Upon completion, the cross sections for the pre-load is automatically detected and indicated in the Canvas as illustrated in Figure 11. Also shown in this figure are the remaining selections, the pre-load stress is 600 Mpa for the duration of 1 second. To create the pre-load, select an appropriate *Name* (say *excavator*), click *Create* and the pre-load object will appear in the list of *Solid Groups* in the tree. You may at any point click on *excavator* and edit the data if mistakes were made. Proceed by creating another pre-load object for the *Solid BOLT* which holds the *front grabhandle* in place, using the same data as for the excavator bolts.

This is a functionality that nicely shows the simplicity we attempt to attain in the Solution Explorer, but we also admit that refinements will be necessary for enhanced flexibility.

3.8.3 Contacts

A vital ingredient in nonlinear implicit simulations is contacts, and the *Contacts* branch offers *Sliding Contact*, *Tied Contact* and *Transducer* as available options. For our model we need to utilize both sliding and tied contacts, a right-click on *Contacts* and selecting will open batch dialogues for the respective types.

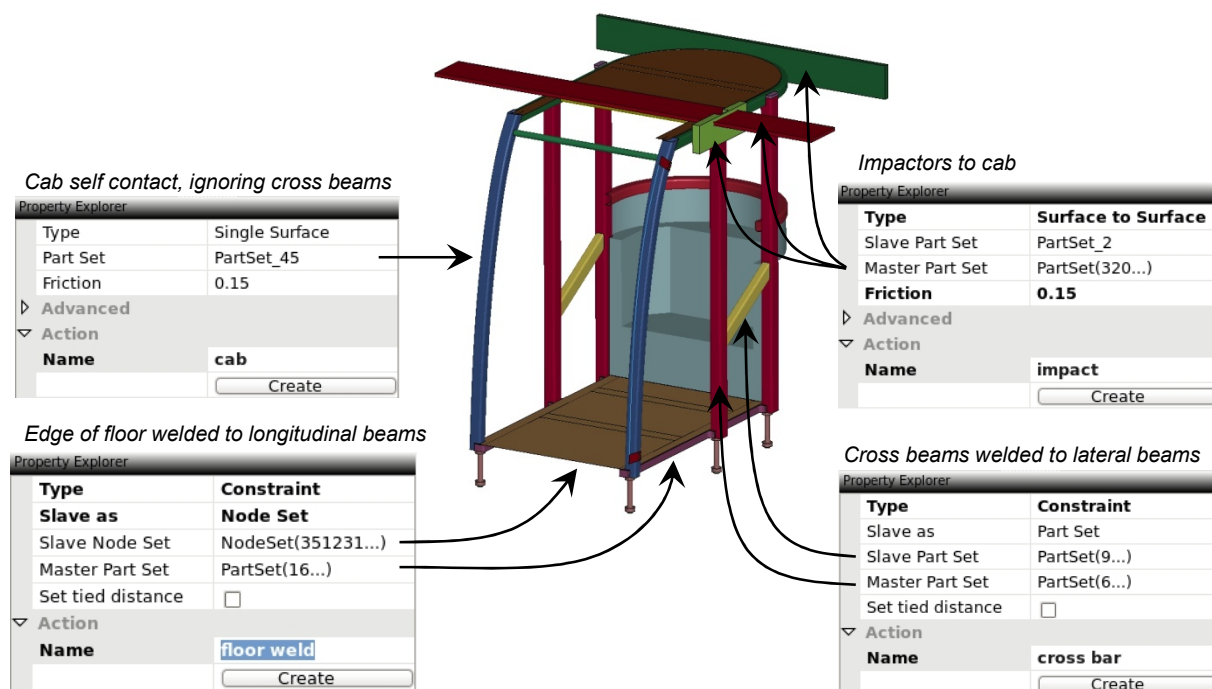


Figure 12 Selected contact definitions

Sliding Contacts

The two sliding contacts we need are a self contact of the cab and a contact between the impactors and the cab. For the first, go to the *Sliding Contacts* dialogue and select *Single Surface* as the *Type*. Then for the *Part Set*, select all parts of the cab but exclude the *crossbeams* since we instead will tie them later. Make sure the three impactors (*ROPS Long*, *LDD Lateral* and *ROPS vertical*) are not part of this set, which they presumably aren't since we deactivated them earlier. Select a *Friction* coefficient of 0.15, *Name* as appropriate and *Create* it. For the impact there is the caveat that we deactivated the impactor parts earlier so they are currently not selectable for creating the contact of interest. The natural thing to do would be to create one contact for each impact case later on, but another option is to temporarily activate the parts and define one contact for all and then deactivate the parts. It does not matter that deactivated parts are members of a contact, the consequence is just that the contact will simply ignore those parts. So, create a *Surface to Surface* contact with the cab as

the *Slave Part Set* and the three impactors as the *Master Part Set*. Just as for the cab self contact, use a *Friction* of 0.15, select an appropriate *Name* and click *Create*. The sliding contact definitions are now completed, see Figure 12 for a recap of our actions.

Tied Contacts

The tied contacts are necessary to maintain the integrity of the cab, we will define a total of six. For each of them in the *Tied Contacts* batch dialogue, the *Type* is a *Constraint* contact and we won't use the option to *Set tied distance* so leave this check box unchecked at all times. For the first contact we will weld the edge of the floor to the longitudinal beams, and for this we need the *Slave as* to be a *Node Set*. For the *Slave Node Set*, select all nodes on the edge of the *floor_plate*, and let the *Master Part Set* be the *floor_solids* (only one part). *Name* it and *Create* to add the first contact. The second contact is used to weld the cross beams to the lateral beams, for this the *Slave as* is a *Part Set*, and the selections of *Slave Part Set* and *Master Part Set* are *crossbeams* and *pillar*, respectively. These first two contacts are illustrated in Figure 12. The remaining tied contacts are created in exact analogy with the second one, and the intention is to tie according to the following table

<i>Slave Part Set</i>	<i>Master Part Set</i>
16 mm bolt	<i>floor_solids</i>
<i>pillar</i> , <i>front_pillars</i>	<i>floor_solids</i>
<i>U_beam</i> floor	<i>floor_solids</i>
<i>pillar</i>	<i>pillar solid cover</i>

which should be fairly straightforward given the exercise of already having created the first two. This completes the contact definitions, and also the *Model* itself.

3.8.4 Uncovered Model topics

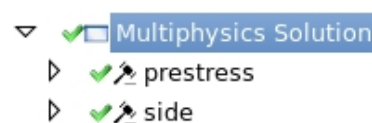
Among the topics not covered in this tutorial we find *Boundary Conditions* and *Connectors*, excluded since they do not apply to the ROPS example. However, we hope that the information provided for the other *Model* content should give a general idea on how to cope also with these undocumented features.

3.9 Output

The *Output* branch is intended for general output, in particular setting the *Time Interval* for output of energy, contact, boundary condition and similar data that should be available for plotting post simulation. Set this interval to 0.01, and the warnings in the tree should disappear. A right-click on *Output* allows you to output detailed *Nodal Data* or *Element Data*, for which you may select the entities of interest. This latter option is not part of this tutorial, and admittedly needs further development to meet expected user demands.

3.10 Remaining cases

At this point, the *prestress* case is complete and you may run it by right-clicking *Multiphysics Solution* and select *Run...* but as stated earlier we intend to proceed with the sequence of impacts, and here we will discuss the creation of these other cases. To this end, right-click on *prestress* and select *Insert Case After*, see Figure 8. This



will clone the *prestress* case to a second case *Case_1* emerging on the same level, and consequently these two trees are identical. Already at this point, rename the case *side* to indicate that this is the side impact case. It is important to stress that the model entities and features *except* boundary and initial conditions are the *same* entities in both cases. For instance is the *crossbeams* part the same part in both cases and the *cab* contact the same contact, and any state variables associated with these two will be transferred from *prestress* to *side* when performing the simulation. The uniqueness of a feature is signified by the *labels*, e.g., a contact that has the same label in two different cases is the same contact and it is not allowed to create a contact with the same label as any other contact. Therefore, a general recommendation when dealing with multiple cases is to complete the first case before continuing with the remaining ones, or otherwise many features common with all cases will require unnecessary extra work. What is left now is to modify the *side* case to do what it is supposed to, and after that repeat the procedure to create the *roof* and *behind* cases. The following briefly states what is needed without elaborating too much on details.

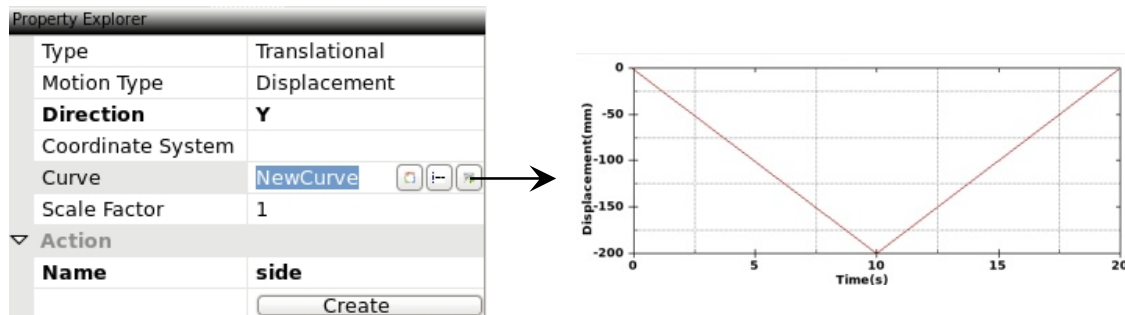


Figure 13 Creating a prescribed motion of the impactor for the side case

3.10.1 side

For the *side* case we operate in the corresponding case tree and the order in which we do things is irrelevant, the following is merely a suggestion.

First, we need to delete the *prestress* objects by right-clicking *Solid Groups* and *Delete*, see Figure 11. This is simply because the pre-loading part is performed in the *prestress* case and of no use here.

Furthermore we need to activate the corresponding impactor by checking the box next to *LDD Lateral* in the list of *Solid Parts*, this will make this part visible in the Canvas. We emphasize that the Canvas will always show the active parts of the *Model* for the case highlighted in the solution tree, so basically the impactor will not show if moving the focus to any item in the *prestress* case tree.

Then we need to prescribe the motion, so right-click on *LDD Lateral* and create a *Prescribed Motion* batch dialogue, see Figure 13. This dialogue demands the *Type* which is set to *Translational*, the *Motion Type* which is set to *Displacement* and the *Direction* which is set to *Y*. The curve defining the motion is edited by clicking the far right icon on *Curve*, and x-y pairs may be entered to form the curve shown in Figure 13. The trough displacement is -200 mm at 10 seconds, and the motion is recovered in 20 seconds.

As the curve data indicates we need to modify the global time frame to make some sense of the case definition, so change the *End Time* and the *Database Plot Interval* on *side* to 20 seconds and 1 second, respectively, revisit Section 3.5 for details.

For the *Analysis* and in particular *Nonlinear Implicit*, see Section 3.6 and 3.7 for reference. We use *Dynamics* as the *Types* on *Integration*, while keeping *Newmark Damped* as the *Methods*. The reason for using dynamics is that the loading condition will presumably induce buckling with potential snap throughs and using statics may not yield a valid solution path. Change *Solution Method* on *Solver* to *Full Newton* for robustness, and use the default values for remaining data. Finally, set *Initial Size* on *Time Step* to 0.2 seconds and use the same value for *Maximum Time Step*. Keep *Aggressive* as the *Change Strategy*, to make sure time steps are recovered after divergence.

These changes complete the *side* case.

3.10.2 roof and behind

The operations needed to construct the *roof* case and the *behind* case correspond to just changing the active parts and deleting/adding boundary conditions accordingly, the time frame and analysis options may be kept as in the *side* impact case. We leave this as an exercise and refer to Section 3.1 and Figure 6 for the relevant description of these cases.

3.11 Running and Post processing

The complete model with all cases defined is run by right-clicking *Multiphysics Solution* and selecting *Run...* This will open LS-Run and you may submit the job according to desired specifications. We assume that you keep Solution Explorer open, or otherwise have re-opened the saved solution file for the project, so the view should be the same as when having finished the model setup. The output files from the LS-DYNA execution will be placed in the directory created for the solution file, see Figure 7.

As soon as these files become available, the Property Editor for selected branches in the Model Tree will be equipped with a *pre* and *post* button, see Section 2.2.4. The *pre* button is intended for model setup and is the mode we have covered so far, while the *post* button is for post-processing data associated with the branch in question and will be covered in the following. For branches that do not have these buttons, there is nothing to post-process and these are always in a model setup (*pre*) mode. Having the same environment for pre- and post-processing is a convenient way of integrating the two since it reduces the body of knowledge required. Experience gained from the model setup comes in handy since data to be post-processed is naturally sought in the correct place, and there is no need to know anything about the underlying output databases. If I for instance want to see the reaction force in the *Body Constraint* of *sleeve*, I would intuitively go to that particular branch, click the *post* button and hope to find what I expect. Likewise, if I want to fringe the stress in the *16 mm bolt*, I would go to that constraint, click *post* and presumably there should be fringing capabilities available. We emphasize that this feature is under construction, and only a few branches are yet equipped with post functionalities.

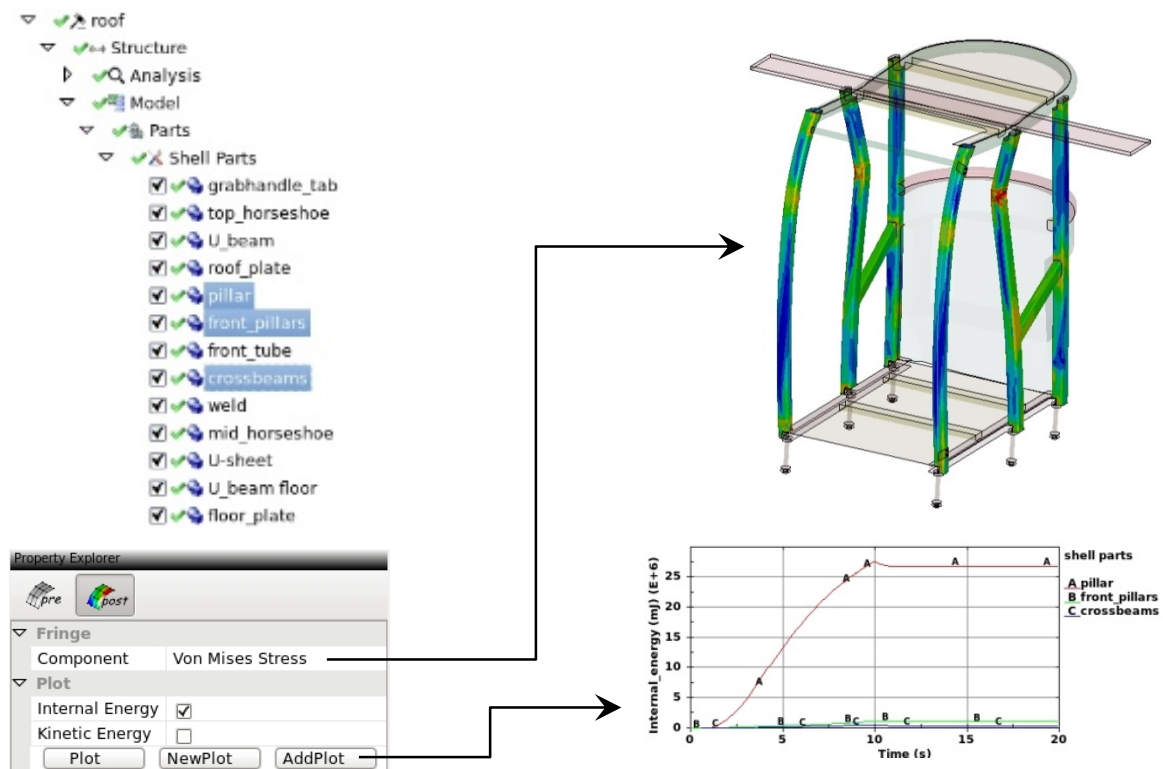


Figure 14 Fringing and plotting components for parts in the roof case

3.11.1 Parts

The post-processing of parts is illustrated in Figure 14, where fringing and plotting are the two capabilities. The way to go about it is straightforward, select the parts of interest as described in 3.8.1 and then fringe or plot the desired quantity. For fringe the selected parts will show in the Canvas with the selected fringe *Component*, the other parts will be transparent and shaded. For plotting you may

<i>Plot</i>	<i>Erase the content of the most recently opened plot window and plot the selected components therein, if no plot window exists then open a new window and plot the selected components therein</i>
<i>NewPlot</i>	<i>Open a new plot window and plot the selected components therein</i>
<i>AddPlot</i>	<i>Keep the content of the most recently opened plot window and plot the selected components therein, if no plot window exists then open a new window and plot the selected</i>

components therein

The number of components to fringe and plot is limited at the moment, more will be added in due time.

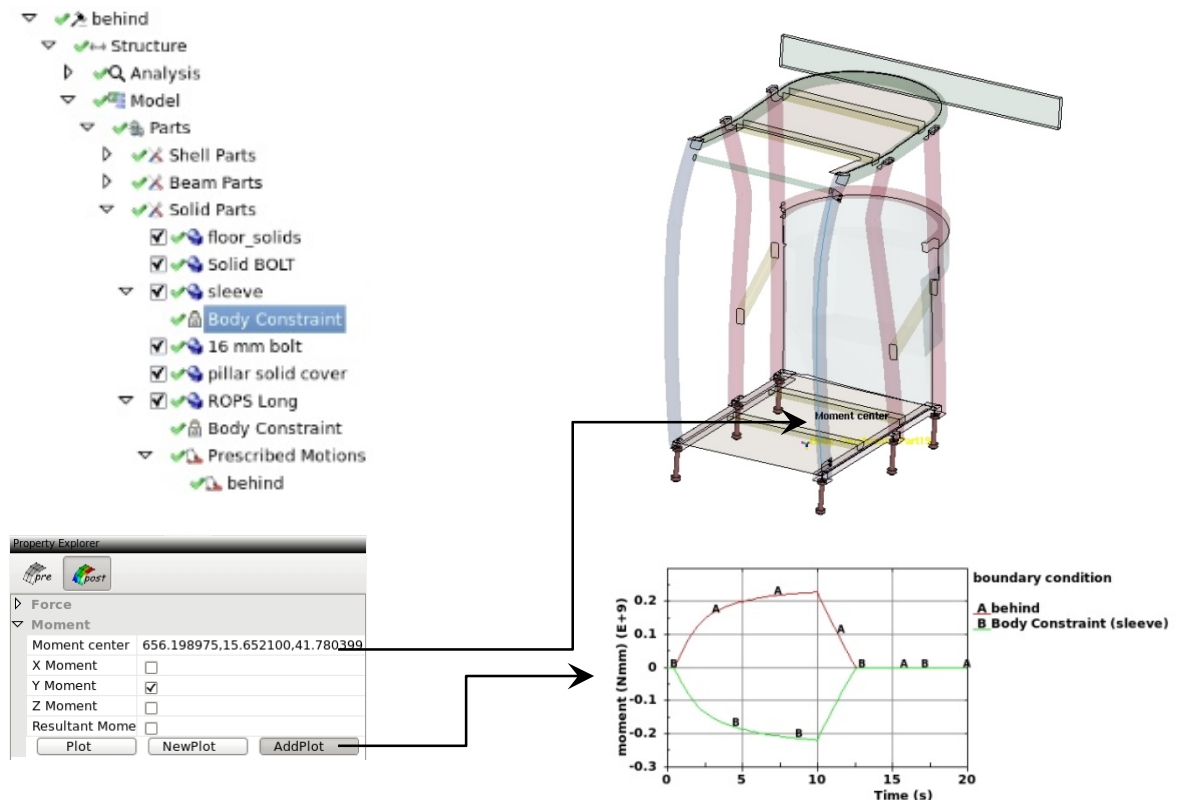


Figure 15 Plotting reaction moments for sleeve and ROPS long in the behind case

3.11.2 Boundary conditions

On various boundary conditions you may plot reaction forces and moments in a fairly straightforward manner. In the list of *Force* or *Moment* components, check the ones to plot and click whatever button that performs the operation desired. The *Moment center* for plotting moments is the center of moment about which the moment is calculated, and this is exemplified in Figure 15. When *Moment center* is highlighted you may pick an arbitrary point in the Canvas, say somewhere on the floor of the cab. Since the only interactions of the cab to its surroundings is through the prescribed motion of the impactor and the constraint to the excavator, these moments with respect to any point in space should balance (if neglecting dynamic effects). Figure 15 plots these two moments in the same graph, and indeed they seem to be of same magnitude and opposite signs. This way of plotting reaction forces will be generically applied to other features as well, for instance, contacts.

4 Summary and Outlook

Simulation in LS-DYNA is about predicting the outcome of real life events involving advanced physical phenomena. A challenge is to make these functionalities available in a format that is easy to understand without waiving the full potential of LS-DYNA. These two objectives are to some extent in conflict and a reasonable trade-off seems a conceivable demand. The Solution Explorer is an attempt to meet this demand, and this paper presents its current state and underlying vision. We believe it is possible to cover a vast amount of applications within a high level graphical environment and the intention is to continue along these lines by complementing the interface with missing and relevant features. Among the topics we need to address is the possibility to populate the solution tree and material database from keyword data, which may become the main focus of development in the near future. Despite not being at the level of maturity we want at the moment, it should still provide main stream users a nice and neat way to set up an implicit thermo-mechanical analysis and we encourage you to try it out and provide feedback to your local LS-DYNA and LS-PrePost distributor.

5 Literature

1. LS-DYNA Keyword User's Manual, Volume I-II, Livermore Software Technology Corporation (LSTC), 2019.
2. <https://en.wikipedia.org/wiki/LS-DYNA>
3. <https://www.computerhistory.org/makesoftware/exhibit/car-crash-simulation>
4. <https://www.dynasupport.com/howtos/implicit/some-guidelines-for-implicit-analyses-using-ls-dyna/ImplicitPackage.zip/view>
5. Appendix P, LS-DYNA Keyword User's Manual, Volume I-II, Livermore Software Technology Corporation (LSTC), 2019.
6. LS-DYNA Theory Manual, Livermore Software Technology Corporation (LSTC), 2019.
7. <ftp://ftp.lstc.com/lstc-winsuite/> (accessible for customers through local distributors)
8. SS-EN ISO 3471:2008