LS-DYNA User's Guide

Ansys
ANSYS, Inc.
Southpointe
2600 Ansys Drive
Canonsburg, PA 15317
ansysinfo@ansys.com
http://www.ansys.com
(T) 724-746-3304
(F) 724-514-9494

Release 2025 R1
January 2025

Published in the U.S.A.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction to LS-DYNA

LS-DYNA is a general-purpose finite element program capable of simulating complex real world problems. It is used by the automobile, aerospace, construction, military, manufacturing, and bioengineering industries. The LS-DYNA solver is optimized for shared and distributed memory Unix, Linux, and Windows based platforms, and it is fully QA'd by Ansys. The code's origins lie in highly nonlinear, transient dynamic finite element analysis using explicit time integration.

"Nonlinear" means at least one (and sometimes all) of the following complications:

- Changing boundary conditions (such as contact between parts that changes over time)
- Large deformations (for example the crumpling of sheet metal parts)
- Nonlinear materials that do not exhibit idealy elastic behavior (for example thermoplastic polymers)

"Transient dynamic" means analyzing high speed, short duration events where inertial forces are important. Typical uses include:

- Automotive crash (deformation of chassis, airbag inflation, seatbelt tensioning)
- Explosions (underwater Naval mine, shaped charges)
- Manufacturing (sheet metal stamping)

LS-DYNA's potential applications are numerous and can be tailored to many fields. In a given simulation, any of LS-DYNA's many features can be combined to model a wide range of physical events. LS-DYNA is one of the most flexible finite element analysis software packages available.

LS-DYNA consists of a single executable file and at the solver level is entirely command line driven. Therefore all that is required to run LS-DYNA is a command shell, the executable, an input file, and enough free disk space to run the calculation. All input files are in simple ASCII format and thus can be prepared using any text editor. Input files can also be prepared within the Ansys Workbench and Ansys Mechanical environments.

There are many third party software products available for preprocessing LS-DYNA input files but the Ansys Workbench LS-DYNA interface combines the perfect combination of power and ease of use. Licensees of LS-DYNA automatically have access to all of the program's capabilities, from simple linear static mechanical analysis up to advanced thermal and flow solving methods.

The LS-DYNA Workbench system takes the power of the LS-DYNA solver and wraps it up into the familiar and easy to use environment of Ansys Workbench and Ansys Mechanical. This unlocks the power of parameterization, CAD import, meshing and all of the other technology which makes Ansys Mechanical the world's number one engineering simulation tool.

Chapter 2: Running LS-DYNA

It is possible to use LS-Run to launch LS-DYNA with a specified input file. But the LS-DYNA system available in Workbench gives you access to the pre- and post-processing capabilities of the Mechanical application to make it easier to set up your LS-DYNA analysis.

Note:

If you are running a version of the LS-DYNA solver earlier than 12.1, you may encounter an issue running LS-DYNA through Workbench on the following Linux versions if the process ID is too large: CentOS 7.9, 8.2, 8.3; Red Hat 7.9, 8.2, 8.3. Before starting Workbench on these versions of Linux, you should issue the command:

```
sysctl -w kernel.pid_max=32768
```

The following sections provide an overview on running LS-DYNA using the Workbench system.

2.1. How to use LS-DYNA in Workbench

The LS-DYNA and LS-DYNA Restart systems are available in the project **Toolbox** in Workbench. To run an LS-DYNA analysis, drag the system into a project and set up and run your model as usual. The LS-DYNA system will create an LS-DYNA keyword (`.k`) file that contains all the necessary information to carry out the analysis, and will run the LS-DYNA solver using that file.

For general information about setting up an Explicit Dynamics analysis, see Explicit Dynamics Workflow.

All the LS-DYNA keywords are implemented according to the LS-DYNA Keyword User's Manual Volumes I, II, III.

All the LS-DYNA keywords that are supported in Workbench are described in detail in Keywords used by LS-DYNA in Workbench (p. 129). Any parameters that are not shown for a card are not used, and their default values will be assigned for them by the LS-DYNA solver.

When using Commands objects with LS-DYNA, be aware of the following:

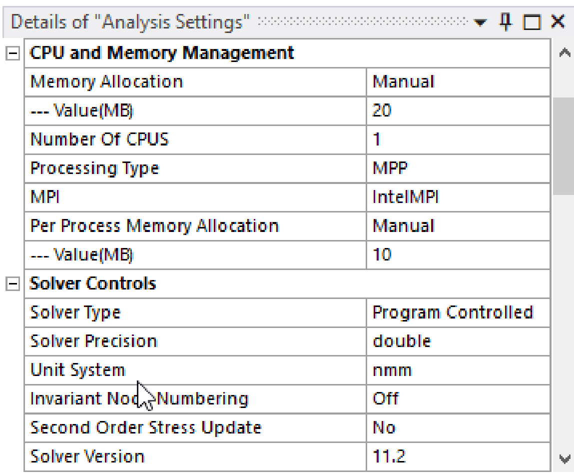
- Keyword cards read from **Commands** object content (renamed to **Keyword Snippets** for LS-DYNA) should not have any trailing empty lines if they are not intentional. This is because some keywords have more than one mandatory card that can be entered as blank lines, in which case the default values for the card will be used. Therefore, trailing blank lines should be used only if intended; otherwise they may cause solver execution errors.

- The first entry in the **Commands** object content must be a command name which is preceded by the * symbol.

2.1.1. Selecting the Version of LS-DYNA to Run

The most current released version of LS-DYNA is installed with your Ansys installation. You may install other versions of LS-DYNA and configure the software so that the other installed versions appear as options in the Analysis Settings (see Exposing Other Versions of LS-DYNA (p. 5)).

2.1.1.1. Analysis Settings



You can select the version of LS-DYNA that you want to run using the fields in the Analysis Settings. The following fields control the selection of the solver version:

Processing Type

Choose whether to use the shared memory parallel processing (SMP), the massively parallel processing (MPP) capabilities of LS-DYNA, or choose Solve Process Settings.

With Solve Process Settings, the processing type is defined by the active setting in the Ribbon toolbar. If "Distributed" is checked, the solution will use MPP. If not, it will use SMP. For more information on Solve Process Settings, see the section Using Solve Process Settings in the Mechanical User's Guide.

If MPP is used, you must also choose the version of **MPI** that you want to use.

Solver Precision

Choose whether to use Single or Double precision floating point representation.

Solver Version

Choose the installed version of LS-DYNA that you want to use. See Exposing Other Versions of LS-DYNA (p. 5) for more information.

In addition to these settings, you can set up the memory allocation and parameters associated with LS-DYNA execution.

Memory Allocation

Select Program Controlled for an automatic calculation of the required memory (MB) for the initialization of the analysis. The estimate is made from the number of nodes in the current model.

Select Manual to set your own memory size. Enter the size of the memory you want to use in the **– Value[MB]** field.

Number of CPUs

Enter the number of CPUs that you want to use to run your job. For an SMP solution, if you enter the number of CPUs as a negative number, the job will run in a manner that will give you consistent results but will take longer.

To use the number of cores defined in the Ribbon toolbar as the Number of CPUs, choose Solve Process Settings for Processing Type and enter 0 for the Number of CPUs. For more information on Solve Process Settings, see the section Using Solve Process Settings in the Mechanical User's Guide.

Per Process Memory Allocation

Select Program Controlled for an automatic calculation of the memory (MB) required for the analysis to run on each node. The same amount of memory will be allocated on each node.

Select Manual to set your own memory size. Enter the size of the memory you want to use in the **– Value[MB]** field.

2.1.1.2. Exposing Other Versions of LS-DYNA

If you have multiple versions of LS-DYNA installed on your system, you can choose to run any of them using the **Solver Version** field in the Analysis Settings. However, you must follow a procedure to allow Workbench to recognize that additional versions are installed.

Each solver `.exe` file should be on your machine and the file `lsdyna_solvers.xml` must be properly defined. This file is created in the folder `C:\Users\<CurrentUser>\AppData\Roaming\Ansys\251\ACTLSDYNA` the first time that Ansys Mechanical is launched and recreated if it happens to be missing on subsequent launches. The content of the xml file before any modification, for a system running on Windows, should look as follows:

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<!--
File containing the default LSDyna solver files
Should contain at least one valid solver to prevent miscallaneous behavior
-->
<LSDynaVersions>
  <Version versionId="11.2">
    <exe os="windows" process="SMP" precision="single"
       path="C:\Program Files\ANSYS Inc\v251\ansys\bin\winx64" name="lsdyna_sp.exe" />
    <exe os="windows" process="SMP" precision="double"
       path="C:\Program Files\ANSYS Inc\v251\ansys\bin\winx64" name="lsdyna_dp.exe" />
    <exe os="windows" process="MPP" mpiType="IntelMPI" precision="single"
       path="C:\Program Files\ANSYS Inc\v251\ansys\bin\winx64" name="lsdyna_mpp_sp_impi.exe" />
    <exe os="windows" process="MPP" mpiType="MSMPI" precision="single"
       path="C:\Program Files\ANSYS Inc\v251\ansys\bin\winx64" name="lsdyna_mpp_sp_msmpi.exe" />
    <exe os="windows" process="MPP" mpiType="IntelMPI" precision="double"
       path="C:\Program Files\ANSYS Inc\v251\ansys\bin\winx64" name="lsdyna_mpp_dp_impi.exe" />
    <exe os="windows" process="MPP" mpiType="MSMPI" precision="double"
```



```
      path="C:\Program Files\ANSYS Inc\v251\ansys\bin\winx64" name="lsdyna_mpp_dp_msmpi.exe" />
  </Version>
</LSDynaVersions>
```

Note:

The individual exe tags are shown on two lines so that they will be more readable. They will be one line in the file.

Every solver file must be uniquely defined by the following attributes:

versionId

Solver version number, such as 11.2

process

Process type, either *SMP* or *MPP*

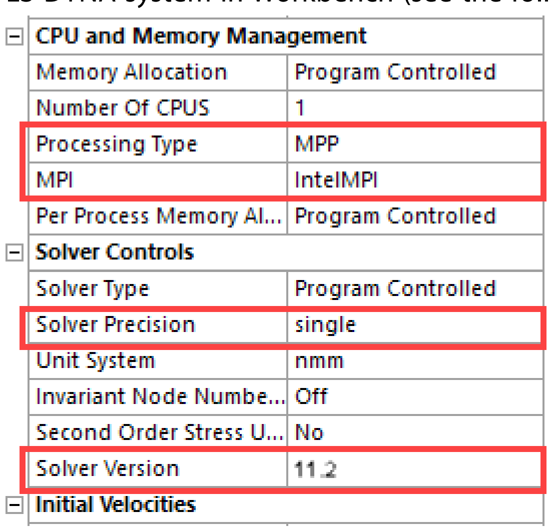
mpiType

The MPI type you want to use, such as IntelMPI (this field is only required when Process = MPP)

precision

The solver precision you want to use, either *single* or *double*

These four attributes are used to create options that appear in the Analysis Settings fields of the LS-DYNA system in Workbench (see the following image).



Additional attributes of the `exe` tag complete the definition of each solver option:

os

Operating system of the machine on which LS-DYNA is running, either *windows* or *linux*

path

Path (relative or absolute) of the folder containing the solver file

name

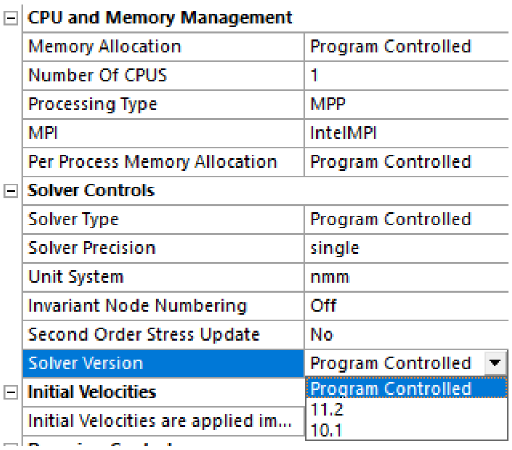
Name of the solver file

If you want to solve with LS-DYNA versions that are not included in the Ansys installation, you need to add a new `Version` tag in the xml file, and one new `exe` tag per solver executable. The attributes `versionId`, `process` (and `mpiType` if the process is "MPP"), `precision`, `os`, `path` and `name` are mandatory. In order for the available options to appear in the UI drop-down menus and the solver files to execute correctly, you must properly define these attributes.

The following example illustrates the necessary modifications to the xml file in order to include the single precision Intel MPI MPP solver from version 10.1.

```
<LSDynaVersions>
  <Version versionId="11.2">
    <exe os="windows" process="SMP" precision="single"
       path="C:\Program Files\ANSYS Inc\v211\ansys\bin\winx64" name="lsdyna_sp.exe" />
    <exe os="windows" process="SMP" precision="double"
       path="C:\Program Files\ANSYS Inc\v211\ansys\bin\winx64" name="lsdyna_dp.exe" />
    <exe os="windows" process="MPP" mpiType="IntelMPI" precision="single"
       path="C:\Program Files\ANSYS Inc\v211\ansys\bin\winx64" name="lsdyna_mpp_sp_impi.exe" />
    <exe os="windows" process="MPP" mpiType="MSMPI" precision="single"
       path="C:\Program Files\ANSYS Inc\v211\ansys\bin\winx64" name="lsdyna_mpp_sp_msmpi.exe" />
    <exe os="windows" process="MPP" mpiType="IntelMPI" precision="double"
       path="C:\Program Files\ANSYS Inc\v211\ansys\bin\winx64" name="lsdyna_mpp_dp_impi.exe" />
    <exe os="windows" process="MPP" mpiType="MSMPI" precision="double"
       path="C:\Program Files\ANSYS Inc\v211\ansys\bin\winx64" name="lsdyna_mpp_dp_msmpi.exe" />
  </Version>
  <Version versionId="10.1">
    <exe os="windows" process="MPP" mpiType="IntelMPI" precision="single"
       path="C:\myLS-DYNAsolvers\v101" name="lsdyna_mpp_sp_impi.exe" />
  </Version>
</LSDynaVersions>
```

After relaunching Mechanical, if you select the fields corresponding to the attributes of the newly defined solver entry, it will appear in the **Solver Version** menu, as in the following figure.



> **Note:**
>
> Solver versions not included with the Ansys product distribution can be installed anywhere you like.

2.1.2. Solving Units

Use the **Unit System** field to set the units that the solver uses. All model inputs will be converted to this set of units during the solve. Results from the analysis will be converted back to the user's unit system in the GUI. For LS-DYNA systems, seven unit systems are available. These unit systems use the following base units:

nmm

Metric (mm, tonne, N, s)

μmks

Metric (μm, Kg, μN, s)

Bft

US Customary (ft, lbm, lbf, s)

Bin

US Customary (in, lbm, lbf, s)

mks

Metric (m, Kg, N, s)

cgs

Metric (cm, g, dyne, s)

mm,ms,kg

Metric (mm, kg, kN, ms)

2.1.3. Explicit-to-Implicit Sequential Solutions

A *sequential solution* is an analysis technique that uses a combination of implicit (general Ansys or LS-DYNA) and explicit (LS-DYNA) solution methods. In problems that require a sequential solution, the results of an explicit analysis are imported into an implicit model (or vice-versa) for the purpose of obtaining a final solution.

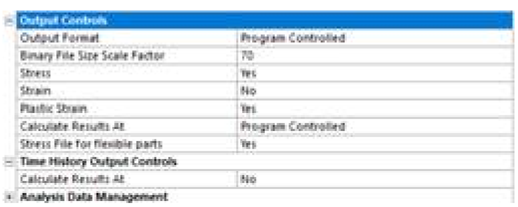
Some of the reasons to perform a sequential solution include:

- Some engineering processes are very complex and contain both dynamic and static phases (for example, initial hoop stress in a pressure vessel before a drop test or the linear elastic springback after sheet metal forming).
- The explicit technique is geared towards solving nonlinear dynamic-impact problems and is not robust for solving static phases of physical phenomena.
- The implicit method is best suited for solving static or quasi-static problems.
- Combining implicit and explicit solvers is an extremely powerful tool for allowing simulations of otherwise intractable engineering problems.

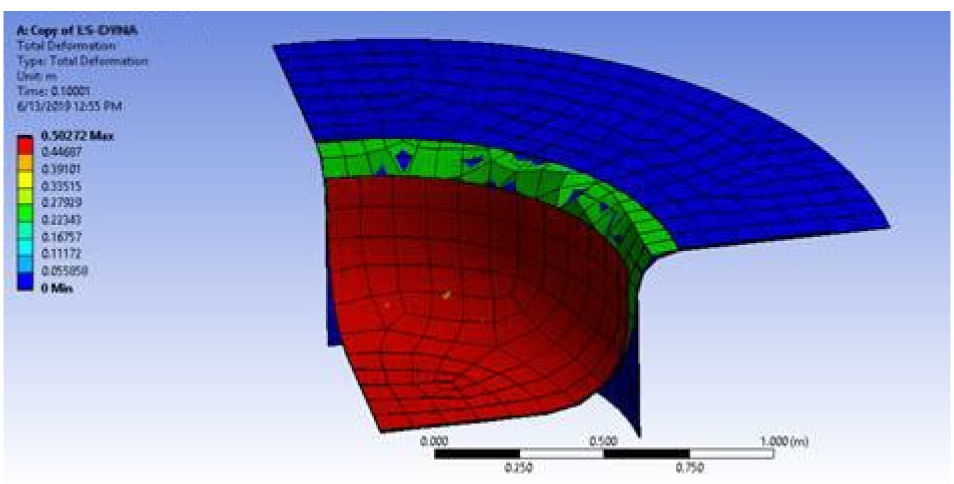
2.1.3.1. Steps to run a Sequential Solution

The basic steps required to perform an explicit-to-implicit sequential solution include:

1. Solve the explicit portion of the analysis. Set the analysis option **Stress File For Flexible Parts** (saving the stresses on deformable parts) to Yes.

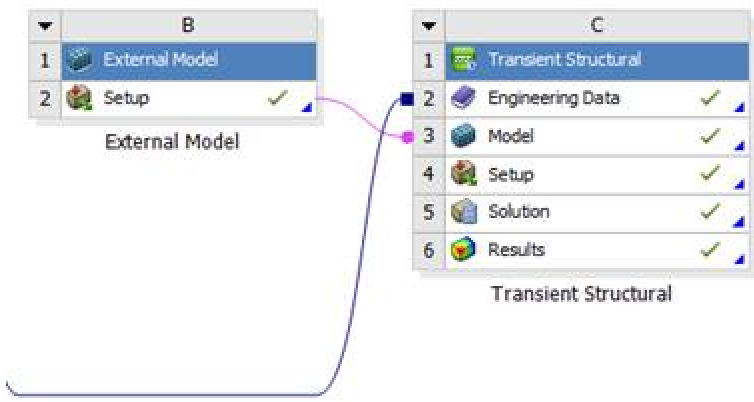


Output Controls	
Output Format	Program Controlled
Binary File Size Scale Factor	70
Stress	Yes
Strain	No
Plastic Strain	Yes
Calculate Results At	Program Controlled
Stress File for flexible parts	Yes
Time History Output Controls	
Calculate Results At	No
Analysis Data Management	

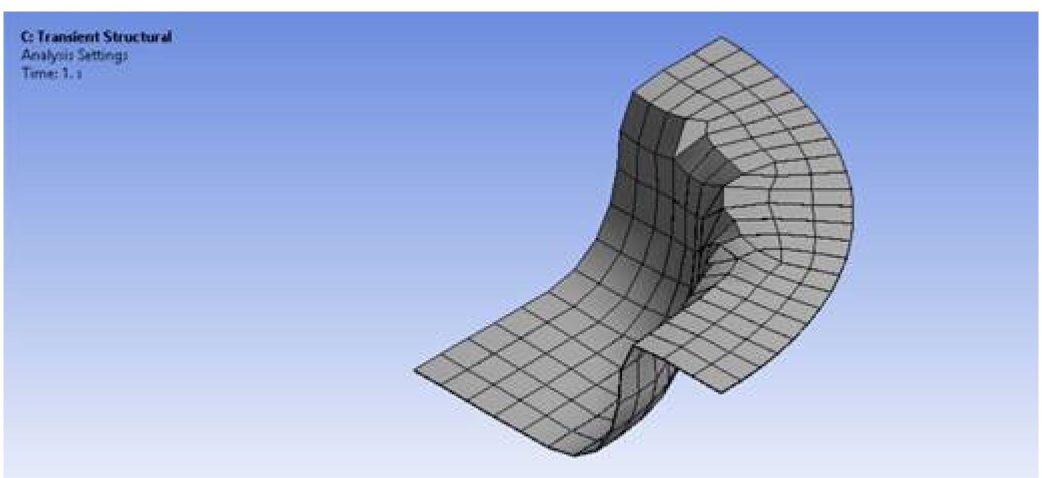


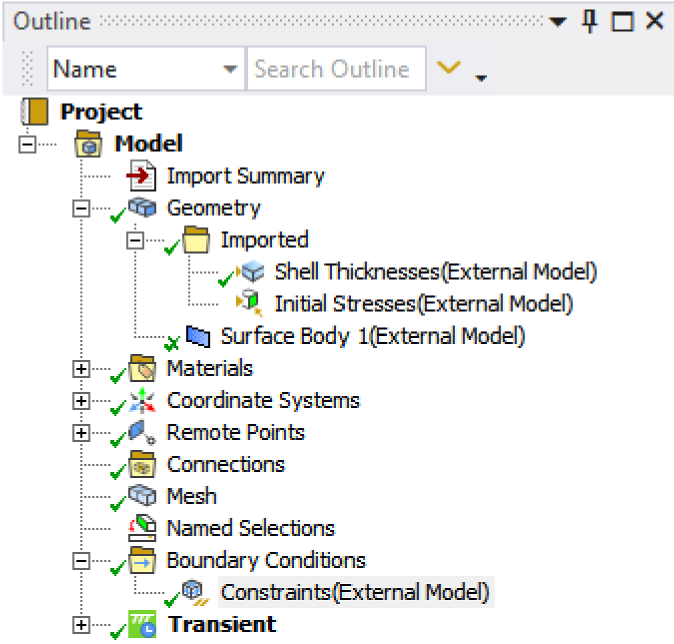
LS-DYNA will create a text file named `dynain`, which contains the deformed geometry along with the stresses on the deformable bodies. Displacements on rigid bodies are not saved in this file.

2. The `dynain` file is an LS-DYNA input file, which can be imported into a Mechanical system using the External Model component system. You must change the file's extension to `.k` before specifying the file in the External Model system. Link the External Model Setup cell to the Model cell of a Static Structural or Transient Structural system, or a LS-DYNA system if the deformed geometry and stresses are meant to be used in a subsequent LS-DYNA analysis.



3. After you open the Mechanical application you see that stresses, shell element thicknesses, and fixed displacement and fixed rotation constraints from the explicit analysis are imported into Mechanical.





4. Redefine the boundary conditions.

5. Solve the new analysis.

Additional Considerations for the LS-DYNA to Mechanical APDL Solver Workflow

If elasto-plastic materials are used in the explicit analysis, compatible elasto-plastic materials must be used in the subsequent Mechanical APDL analysis.

If the explicit analysis is using elasto-plastic materials, a minimum of five integration points should be used on shell elements if the subsequent analysis uses the Mechanical APDL solver. Additionally, an odd number of integration points for shell elements should be used in this workflow.

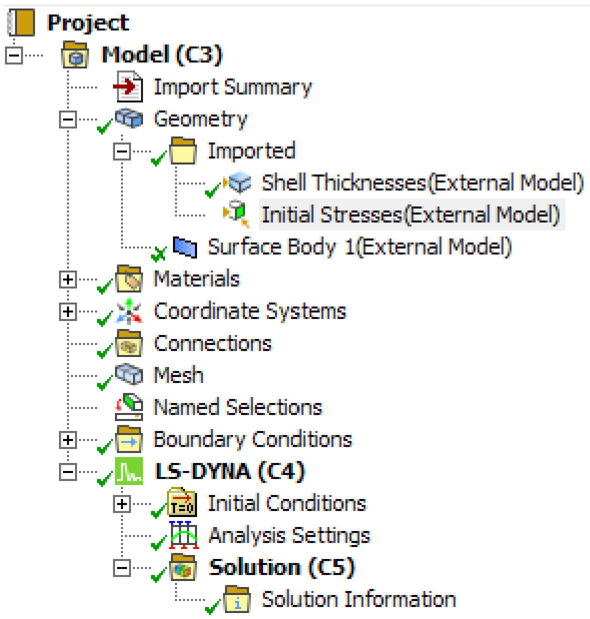
Limitations

The Mechanical APDL analysis relies on the initial state. Initial state does not support shear transfer on shell elements. If explicit stresses in the ZX and the YZ direction in the Mechanical APDL element coordinate system are significant, using this workflow is not recommended as Mechanical APDL will set the YZ and XZ shear to zero in all elements in their respective elemental coordinate system, regardless of the values imported from the `.k` file.

Mechanical APDL calculates the middle layer stresses using the top and bottom stresses by default. This can result in a difference in results. You can insert command snippets to import the value of the middle layer stresses from the external model file if necessary.

2.1.4. Using LS-DYNA Implicit Features for the Implicit Calculation

LS-DYNA Implicit features can also be used as an alternative to the Mechanical APDL solver, in the Implicit part of the workflow. The main benefit is that most of the LS-DYNA materials models are also available, and usable in the Implicit calculation.



To use LS-DYNA implicit features, set the option **Explicit Solution Only** to No.

Step Controls	
End Time	1
Time Step Safety Factor	0.9
Maximum Number Of Cycles	10000000
Automatic Mass Scaling	No
CPU and Memory Management	
Memory Allocation	Program Controlled
Number Of CPUS	1
Processing Type	Program Controlled
Solver Controls	
Solver Type	Program Controlled
Solver Precision	Program Controlled
Unit System	nmm
Explicit Solution Only	No

Then set the following options:

Implicit Controls	
Type	Implicit
Initial Time Step	0.1
Line Search	0
Displacement Convergence	0
Stabilization	On
---Scale Factor	0.001
Start Time	0
End Time	0

- **Initial Time Step**: Initial time step size for implicit analysis

- **Line Search**: Line search can be useful for enhancing convergence, but it can be expensive (especially with plasticity). You might consider setting Line Search on in the following cases:
  - When your structure is force-loaded (as opposed to displacement-controlled).
  - If you are analyzing a "flimsy" structure which exhibits increasing stiffness (such as a fishing pole).
  - If you notice (from the program output messages) oscillatory convergence patterns.
- **Displacement Convergence**: Displacement convergence tolerance. This criteria controls the convergence on displacements. When the displacement norm ratio is below the displacement convergence, equilibrium is achieved. Small values will lead to a more accurate solution, at the expense of more iterations.
- **Stabilization**: Convergence difficulty can happen when importing stresses from an explicit analysis into an implicit analysis. Nonlinear stabilization techniques can help achieve convergence. Nonlinear stabilization can be thought of as adding artificial dampers to all of the nodes in the system. Any degree of freedom that tends to be unstable has a large displacement causing a large damping/stabilization force. This force reduces displacements at the degree of freedom so that stabilization can be achieved.

  There are two options for controlling stabilization:
  - Off (default): Deactivates stabilization.
  - On: Specifies stabilization should be used.
    - **Scale Factor**: Enter the scale factor value that the LS-DYNA solver uses to calculate stabilization forces.
    - **Start Time**: Enter the start time for the stabilization. Defaults to the beginning of the implicit calculation.
    - **End Time**: Enter the end time for the stabilization.

2.2. Licensing Requirements

The LS-DYNA environment is available to all customers with an Ansys LS-DYNA license and requires that license to run analyses.

The LS-DYNA environment is available for pre- and post-processing (only) with the Enterprise Prepost license.

SMP and MPP Parallel processing are both available. They do, however, require the use of Ansys LS-DYNA HPC licenses. The standard Ansys HPC packs and HPC Workgroup licenses do not work with LS-DYNA. Use Ansys LS-DYNA HPC licenses for running in parallel.

2.3. Running a Distributed Solution

For large models, you can use the Shared Memory Parallel processing (SMP) or the Massively Parallel Processing (MPP) capabilities of LS-DYNA to shorten the elapsed time necessary to run an analysis. To use either of these features, you must purchase the appropriate number of Ansys LS-DYNA HPC licenses.

The HPC license incorporates both SMP and MPP capabilities. Please contact your Ansys sales representative for more information on purchasing the appropriate licenses.

2.3.1. Shared Memory Parallel Processing

The shared memory parallel processing capabilities allow you to distribute model-solving power over multiple processors on the same machine. To use this feature you must have a machine with at least many cores in the computer as the number of LS-DYNA processes, and you must have an Ansys LS-DYNA HPC license for each process beyond the first one.

When you are using shared memory parallel processing, the calculations may be executed in different order, depending on CPU availability and the workload on each CPU. Because of this, you may see slight differences in the results when you run the same job multiple times. To avoid these differences, you can specify the number of CPUs as a negative number to maintain consistency. Maintaining consistency can result in an increase of up to 15% in CPU time.

2.3.2. Massively Parallel Processing

The massively parallel processing (MPP) capabilities of LS-DYNA allow you to run the LS-DYNA solver over a cluster of machines or use multiple processors on a single machine. To use the LS-DYNA MPP feature, you must have an Ansys LS-DYNA HPC license for each processor beyond the first one.

Before running an analysis using LS-DYNA MPP, you must have supported MPI software correctly installed, and the machines running LS-DYNA MPP must be properly configured.

Table 2.1: LS-DYNA MPP MPI Support on Windows and Linux

MPI version for DYNA MPP	64-bit Windows	64-bit Linux
Intel MPI	X	X
MS MPI	X	n/a

2.3.3. Configuring LS-DYNA in Parallel

To run LS-DYNA in parallel on a single machine, no additional setup is required.

To run an analysis with LS-DYNA in parallel on a cluster, some configuration is required as described in the following sections:

2.3.3.1. Prerequisites for Running LS-DYNA in Parallel

If you are running on a single machine, there are no additional requirements for running a distributed solution.

If you are running across multiple machines (for example, a cluster), your system must meet these additional requirements to run a distributed solution.

- Homogeneous network: All machines in the cluster must be the same type, OS level, chip set, and interconnects.
- You must be able to remotely log in to all machines, and all machines in the cluster must have identical directory structures (including the Ansys 2025 R1 installation, MPI installation, and working directories). Do not change or rename directories after you've launched LS-DYNA.
- All machines in the cluster must have Ansys 2025 R1 installed, or must have an NFS mount to the Ansys 2025 R1 installation. If not installed on a shared file system, Ansys 2025 R1 must be installed in the same directory path on all systems.
- All machines must have the same version of MPI software installed and running. Table 2.2: Platforms and MPI Software (p. 15) shows the MPI software and version level supported for each platform.

2.3.3.2. MPI Software

The MPI software supported by LS-DYNA in parallel depends on the platform (see Table 2.2: Platforms and MPI Software (p. 15)).

The files needed to run LS-DYNA in parallel using Intel MPI are included on the installation media and are installed automatically when you install Ansys 2025 R1. Therefore, when running on a single machine (for example, a laptop, a workstation, or a single compute node of a cluster) on Windows or Linux, or when running on a Linux cluster, no additional software is needed. However, when running on multiple Windows machines you must use a cluster setup, and you must install the MPI software separately as described later in this section.

Table 2.2: Platforms and MPI Software

Platform	MPI Software
Linux	Intel MPI 2018.3.222
Windows 10 (single machine)	Intel MPI 2018.3.210[a]
Windows Server 2016 (cluster)	Microsoft HPC Pack (MS MPI v10.1.12)

[a] MS MPI is an alternative to Intel MPI for a single machine on Windows.

2.3.3.3. Installing the Software

Install Ansys 2025 R1 following the instructions in the *Ansys, Inc. Installation Guides* for your platform. Be sure to complete the installation, including all required post-installation procedures.

To run LS-DYNA in parallel on a cluster, you must:

- Install Ansys 2025 R1 on all machines in the cluster, in the exact same location on each machine.
- For Windows, you can use shared drives and symbolic links. Install Ansys 2025 R1 on one Windows machine (for example, `C:\Program Files\ANSYS Inc\V251`) and then *share* that installation folder. On the other machines in the cluster, create a symbolic link (at `C:\Program Files\ANSYS Inc\V251`) that points to the UNC path for the shared folder. On Windows systems, you must use the Universal Naming Convention (UNC) for all file and path names for LS-DYNA in parallel to work correctly.

- For Linux, you can use the exported NFS file systems. Install Ansys 2025 R1 on one Linux machine (for example, at `/ansys_inc/v251`), and then export this directory. On the other machines in the cluster, create an NFS mount from the first machine to the same local directory (`/ansys_inc/v251`).

Installing MPI software on Windows

You can install Intel MPI from the installation launcher by choosing **Install MPI for Ansys Parallel Processing**. For installation instructions see Intel-MPI 2021.8.0 Installation Instructions in the *Ansys, Inc. Installation Guides*.

Microsoft HPC Pack (Windows HPC Server 2016)

You must complete certain post-installation steps before running LS-DYNA in parallel on a Microsoft HPC Server 2016 system. The post-installation instructions provided below assume that Microsoft HPC Server 2016 and Microsoft HPC Pack (which includes MS MPI) are already installed on your system. The post-installation instructions can be found in the following README files:

```
Program Files\ANSYS Inc\V251\tp\MPI\WindowsHPC\README.mht
```

or

```
Program Files\ANSYS Inc\V251\tp\MPI\WindowsHPC\README.docx
```

The user must be a registred user on the HPC cluster.

"Client utilities" from Microsoft HPC Pack must be installed on the computer which submits the job. Use the same version of HPC Pack as is used on the HPC cluster.

Store the credentials for submitting to the cluster by running this command:

```
hpccred setcreds /user:MYDOMAIN\myusername /scheduler:myhpcserver
```

in a command prompt after substituting MYDOMAIN, myusername and myhpcserver.

The input and solver files have to be accessible on the Compute Nodes on the cluster. This typically means that the input and solver files should be placed on a disk share, specified with their UNC paths; in other words starting with \\FILESERVER\.

Chapter 3: Workflow

The version of LS-DYNA in the Ansys installation is able to run the complete keyword set published in the LS-DYNA Keyword User's Manual Volumes I, II, III. In the current implementation, a subset of the extensive list of keyword inputs can be generated by the LS-DYNA Workbench system. The following areas of use are not supported for the LS-DYNA Workbench system:

- User-defined material definition
- LS-DYNA user subroutines
- Mesh-free methods: EFG
- Some special elements such as Seat Belt and others
- Fluid Structure Interaction (FSI)

Advanced, knowledgeable users may modify the input file generated and take advantage of the full keyword set. Note that post-processing may not work correctly with modified `.k` files. For those cases use LS-Prepost, available in the standard installation.

This section describes all of the steps needed to run an analysis using the LS-DYNA system.

3.1. Setting up a Project

The general guidelines for setting up an explicit dynamics analysis can be found in Explicit Dynamics Workflow. LS-DYNA related information is included in that chapter. The use of the LS-DYNA system is described in this section.

3.1.1. Defining Materials

You can find information about Materials in Define Engineering Data.

Note:

- The Material Assignment folder is not supported by LS-DYNA.
- In Engineering Data, temperatures defined in the Material Field Variable section are not used by the LS-DYNA analysis system. In general coefficients that are temperature dependent are not taken into account. Only the first value in the table is used for these coefficients, and you can only have one reference temperature even if multiple materials are used in the model.

3.1.2. Attaching Geometry

You can find information about Geometry in Attach Geometry.

Note:

LS-DYNA cross sections are not fully compatible with cross sections available in Design Modeler. When using a LS-DYNA system the Z, Hat ,and Channel cross sections are exported with the following limitations.

- For the Z cross section, the LS-DYNA type 6 is used. It imposes the following restrictions:

  W1 = W2 (W2 is assumed equal to W1)

  t1 = t2 (t1 is assumed equal to t2)

- For the Hats cross section, the LS-DYNA type 21 is used. It imposes the following restrictions:

  W1 = W2 (W2 is assumed equal to W1)

  t1 = t2 = t3 = t4 = t5

- For the Channel cross section, the LS-DYNA type 2 is used. It imposes the following restrictions:

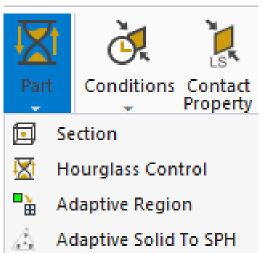
  W1 = W2 (W2 is assumed equal to W1)

  t1 = t2 (all the thicknesses are assumed identical)

3.1.3. Defining Part Behavior

You can find information about Part Behavior in Define Part Behavior.

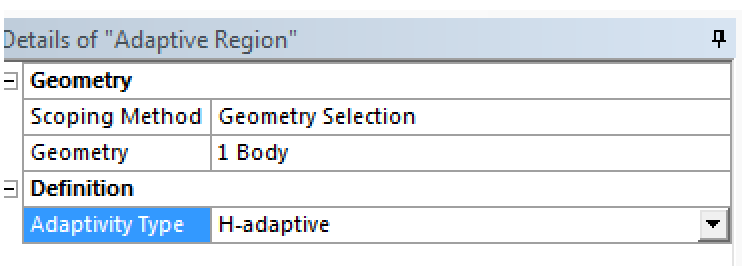
Add the objects under **Part** from the LSDYNA Pre tab to specify **Section** properties such as element formulations, and specify different **Hourglass Control** objects for each part. You can also add an Adaptive Region or an Adaptive Solid to SPH object to your model.



Note:

LS-DYNA supports both Full Mesh and Dimensionally Reduced Behavior for rigid body meshing.

3.1.3.1. Adaptive Region



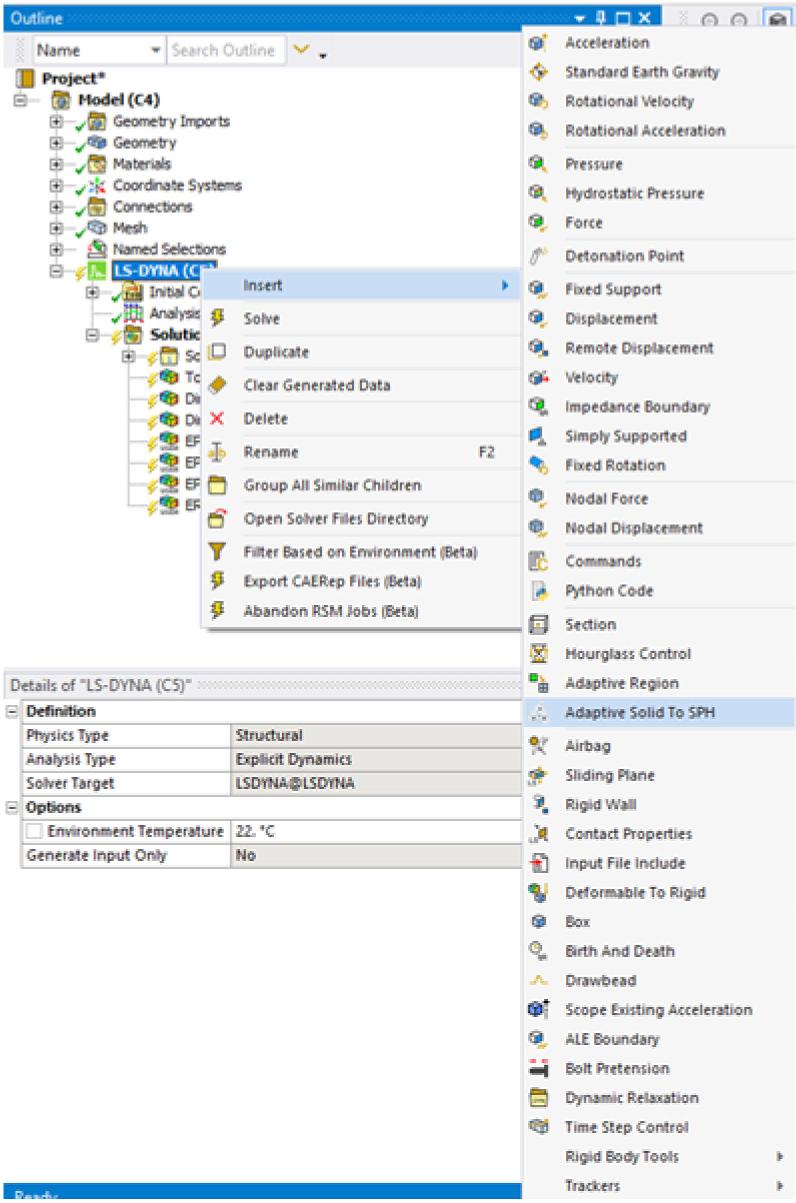
In metal forming and high-speed impact analyses, a body may experience very large amounts of plastic deformation. Single point integration explicit elements, which are usually robust for large deformations, may give inaccurate results in these situations due to inadequate element aspect ratios. To counteract this problem, LS-DYNA has the ability to automatically remesh a surface during an analysis to improve its integrity. This capability, known as adaptive meshing, is controlled with the adaptive region:

- -adaptive for 3-D shells.
- Passive -adaptive for 3-D shells. The elements in this part will not be split unless their neighboring elements in other parts need to be split more than one level.

3.1.3.2. Adaptive Solid to SPH

Adaptive Solid to SPH creates SPH particles to either replace or supplement solid Lagrangian elements. Applications of this feature include adaptively transforming a Lagrangian solid part to SPH particles when the Lagrangian solid elements comprising those parts fail. One or more SPH particles (elements) will be generated for each failed element. The SPH particles replacing the failed solid Lagrangian elements inherit all the Lagrange nodal quantities (like displacement, velocity and acceleration) and all the Lagrange integration point quantities (like stress and strain) of these failed solid elements. Those properties are assigned to the newly activated SPH particles. The newly created SPH part can have different material properties using the **Material Assignment** field.

To insert an **Adaptive Solid to SPH** object, right-click **LS-DYNA** and select **Insert** → **Adaptive Solid to SPH** as depicted in the following image.



The input fields in the **Details** panel of Adaptive Solid to SPH are shown here.

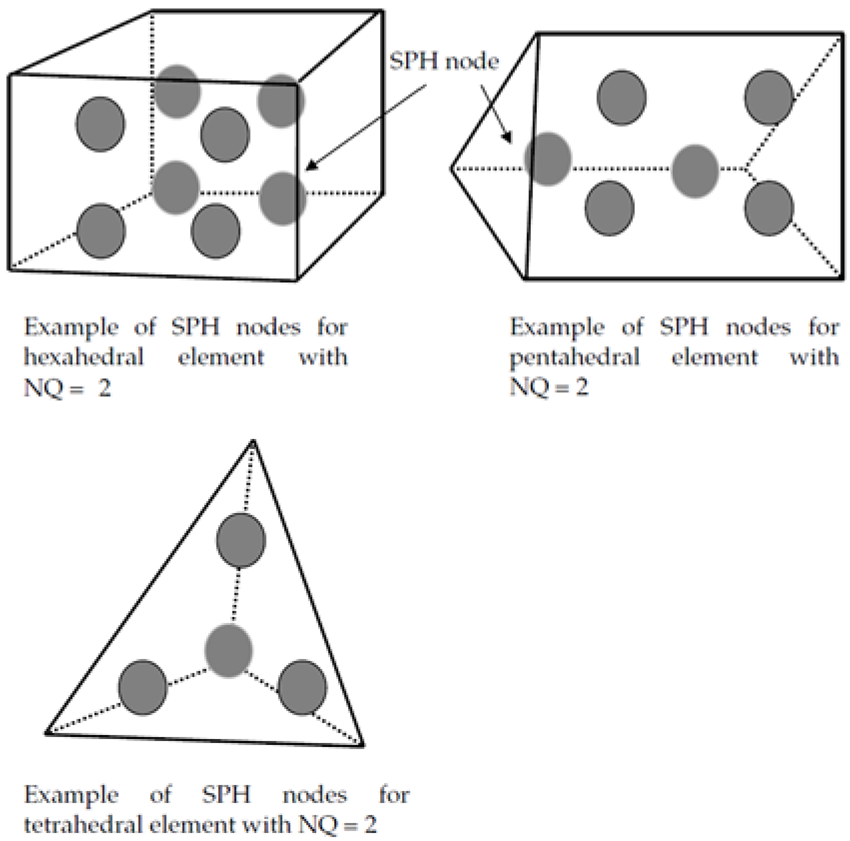
Details of "Adaptive Solid To SPH"

Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Per Element Direction number of particles	1
Coupling Type	Debris
Coupling Start	From Beginning
Material Assignment	User Defined
Material	mat1
Section SPH controls	
Smoothing Length Constant	1.2
Maximum Scale Factor	2
Minimum Scale Factor	0.2

You must scope this object to a solid part having a material compatible with SPH. Shell and beam elements are not allowed for this object. If you scope an object with an unsuitable material or geometry type, a warning message will be issued.

Adaptive Solid to SPH uses the keyword *DEFINE_ADAPTIVE_SOLID_TO_SPH.

Per Element Direction number of particles is the number of SPH particles generated with respect to each direction of the solid element. This field sets the NQ parameter of the *DEFINE_ADAPTIVE_SOLID_TO_SPH keyword. For instance, when using hexahedral elements, the number of SPH particles generated inside a solid element for NQ=2 is 2*2*2=8 particles. For tetrahedral and pentahedral elements the number of particles is described in the following image.



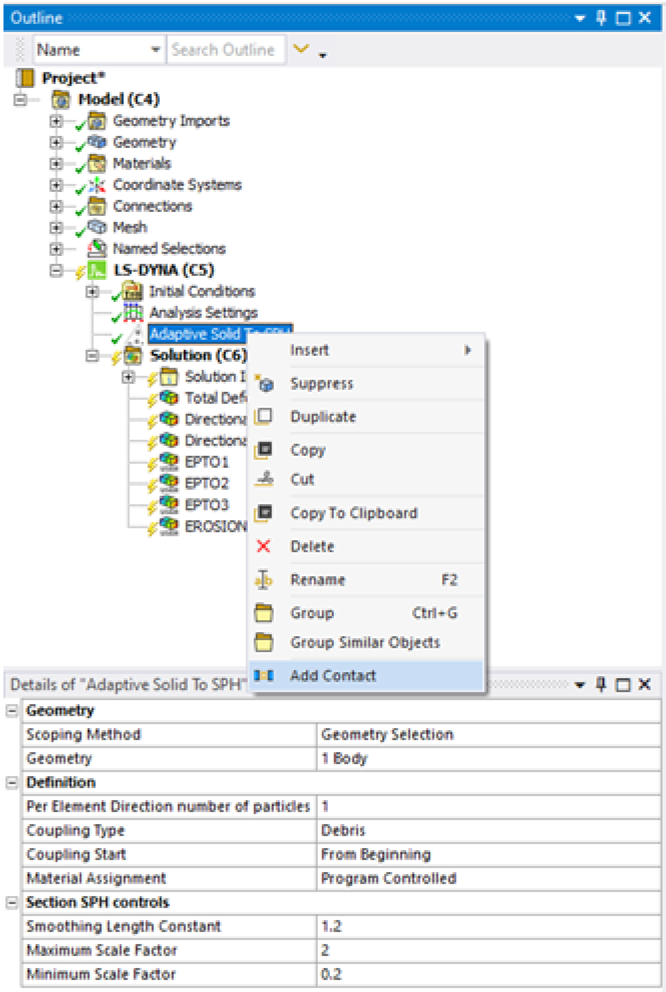
Two options for the **Coupling Type** are available: Debris, and Coupled to Solid Element. This field is used for the ICPL parameter of the *DEFINE_ADAPTIVE_SOLID_TO_SPH keyword.

Two options for the **Coupling Start** are available: From Beginning, or When Solid Element Fails. This field is used for the IOPT parameter of the *DEFINE_ADAPTIVE_SOLID_TO_SPH keyword.

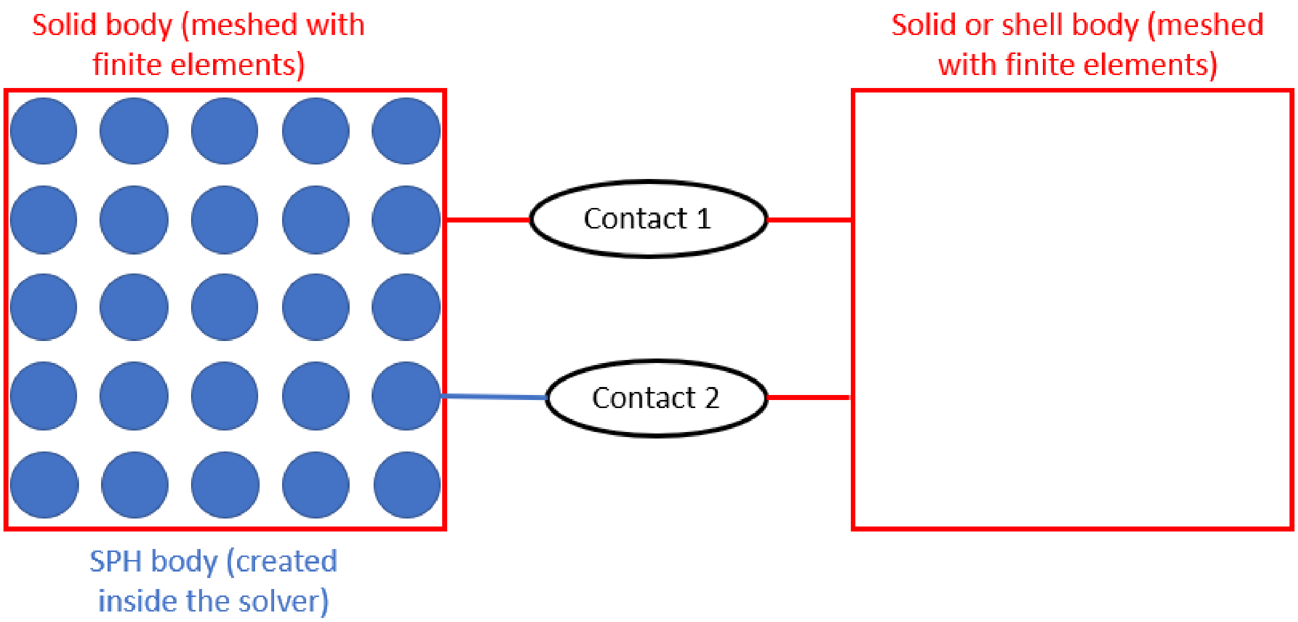
The newly created SPH part can have different material properties by setting **Material Assignment** in the Adaptive Solid to SPH object to User Defined, as shown in the previous Details panel image. All materials defined in Engineering Data are available in the **Material** field. When Material Assignment is set to Program Controlled, the SPH part will have the same material as the solid part.

For the SPH part created inside the solid part, the following Section fields are available and can be modified: **Smoothing Length Constant**, **Maximum Scale Factor**, and **Minimum Scale Factor**.

The SPH part is created inside the solver and is not visible in the Mechanical UI. Therefore, it can't be selected from the model tree if you want to scope the SPH part in a contact. To do this, add a new contact object under the Adaptive Solid to SPH object as shown in the following image. The contact object is inserted only when Material Assignment is set to User Defined.



This contact object allows you to define a contact between the SPH part created inside the solver (Contact Body) and Lagrangian parts (Target Bodies) that are visible in Mechanical as illustrated by the following figure.



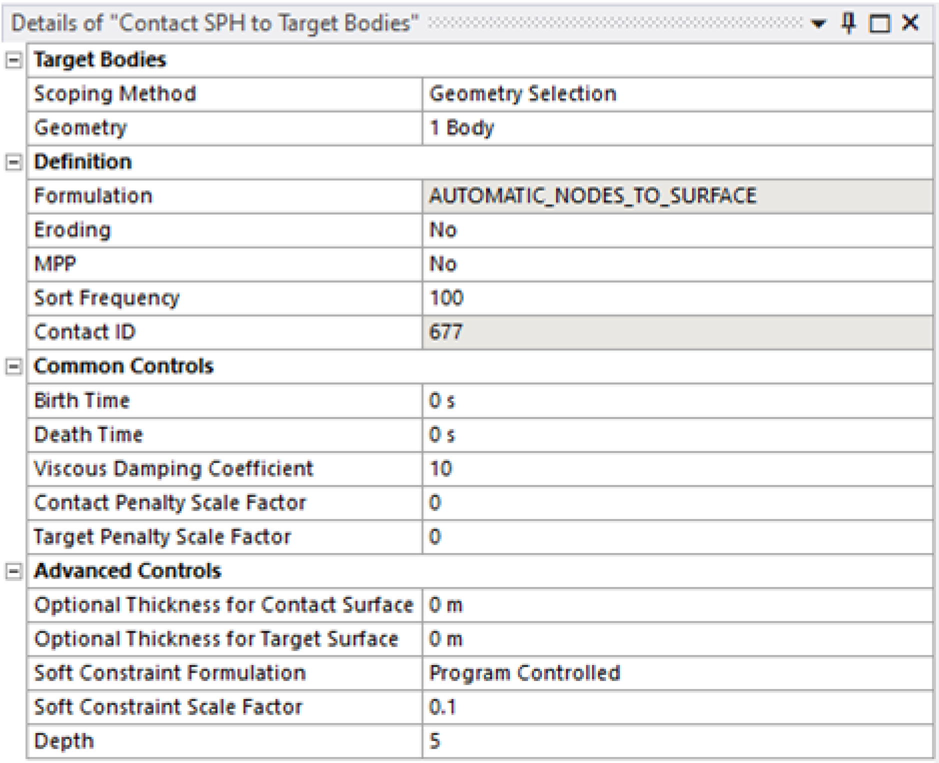
In the previous figure, Contact 1 is a surface-to-surface contact that can be defined with the manual contact region object. Contact 2 is a node-to-surface contact linking the inner solver SPH Part (Contact Body) to the segments of the Target Bodies (solid or shell).

For the case of an SMP solution, the contact object properties would look as follows:

Details of "Contact SPH to Target Bodies"

Target Bodies	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Formulation	AUTOMATIC_NODES_TO_SURFACE
Eroding	No
Sort Frequency	100
Contact ID	677
Common Controls	
Birth Time	0 s
Death Time	0 s
Viscous Damping Coefficient	10
Contact Penalty Scale Factor	0
Target Penalty Scale Factor	0
Advanced Controls	
Optional Thickness for Contact Surface	0 m
Optional Thickness for Target Surface	0 m
Soft Constraint Formulation	Program Controlled
Soft Constraint Scale Factor	0.1
Depth	5

The following figure shows the contact properties when MPP is enabled for the analysis. You can choose not to use MPP for the contact calculation. Not using MPP for contact is sometimes more efficient.



Details of "Contact SPH to Target Bodies"

Target Bodies	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Formulation	AUTOMATIC_NODES_TO_SURFACE
Eroding	No
MPP	No
Sort Frequency	100
Contact ID	677
Common Controls	
Birth Time	0 s
Death Time	0 s
Viscous Damping Coefficient	10
Contact Penalty Scale Factor	0
Target Penalty Scale Factor	0
Advanced Controls	
Optional Thickness for Contact Surface	0 m
Optional Thickness for Target Surface	0 m
Soft Constraint Formulation	Program Controlled
Soft Constraint Scale Factor	0.1
Depth	5

When Eroding is set to Yes, an additional block of input values appears as shown in the following figure:

Details of "Contact SPH to Target Bodies"

Target Bodies	
Scoping Method	Named Selection
Named Selection	2Plaques
Definition	
Formulation	ERODING_NODES_TO_SURFACE
Eroding	Yes
MPP	No
Sort Frequency	100
Contact ID	1003
Common Controls	
Birth Time	0 s
Death Time	0 s
Viscous Damping Coefficient	10
Contact Penalty Scale Factor	1E-12
Target Penalty Scale Factor	1E-12
Advanced Controls	
Optional Thickness for Contact Surface	0 m
Optional Thickness for Target Surface	0 m
Soft Constraint Formulation	Program Controlled
Soft Constraint Scale Factor	0.1
Depth	5
Eroding Controls	
Symmetry Plane Option	Program Controlled
Erosion Interior Node Option	Program Controlled
Solid Elements Treatment	Program Controlled

You must define the target bodies for the contact. Two scoping methods are supported: Geometry Selection and Named Selection. Only solid and surface geometry selections are supported.

If you set **Eroding** to Yes, the card *CONTACT_ERODING_NODES_TO_SURFACE is written. When the property Eroding is set to No, the card *CONTACT_AUTOMATIC_NODES_TO_SURFACE is written.

The sort frequency corresponds to the parameter BSORT of Optional Card A in the case of SMP processing, and to the parameter BCKT of MPP Card 1 in the case of MPP processing.

Common Controls, Advanced Controls and Eroding Controls parameters are identical to the Contact Property object, and its definition can be found in Keywords Created from the Contact Properties Object (p. 201).

This contact supports only Penalty and Soft Constraint formulations. The recommended formulation for an SPH simulation is the soft constraint (SOFT=1) and is used by default (Program Controlled).

When Material Assignment is set to Program Controlled and the scoped solid in the Adaptive Solid to SPH object is used in a manual contact region, a node-to-surface contact card is written between the SPH part (created inside the solver) and the target bodies. In other words, when Contact 1 (in the earlier figure) is defined, Contact 2 is written implicitly in the input file.

To view results on the created SPH particles, see Post-processing for Solver-Created SPH Elements (p. 66).

3.1.3.3. ISPH Region

The Incompressible SPH (ISPH) solver allows you to solve FSI problems that involve Structural and Fluid bodies. This ISPH Region allows you to set the material properties for fluid and solid bodies. In addition, for solid bodies, the ISPH Region will mesh the interaction surface of the body with SPH particles and you can set the space between the SPH particles.

To set the material properties of the fluid part you must scope an ISPH Region object to the body, set the **Type** to fluid, and enter the material properties of the fluid.

Details of "ISPH Region Water"

Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Type	Fluid
ISPH Fluid Material Controls	
Density	2.1 kg/m^3
Dynamic Viscosity	2.2 Pa·s
Numerical Surface Tension 1	2.3 m/s^2
Numerical Surface Tension 2	2.4 m/s^2

If a solid structural body is used in the model, you must extract its skin (using a modeling program like Discovery Modeling) and define it as a new rigid shell body. The shell body should be meshed with triangles, then an ISPH Region should be scoped to the shell body. The interaction surface of the shell body is meshed with SPH particles using the ISPH Region. The ISPH Region will discretize the surface to SPH particles within the solver. Set the **Type** of the ISPH Region to Solid. You can then set the material properties for the solid, the space between the SPH elements, and whether you want to output the mesh to a file.

Details of "ISPH Region Car"

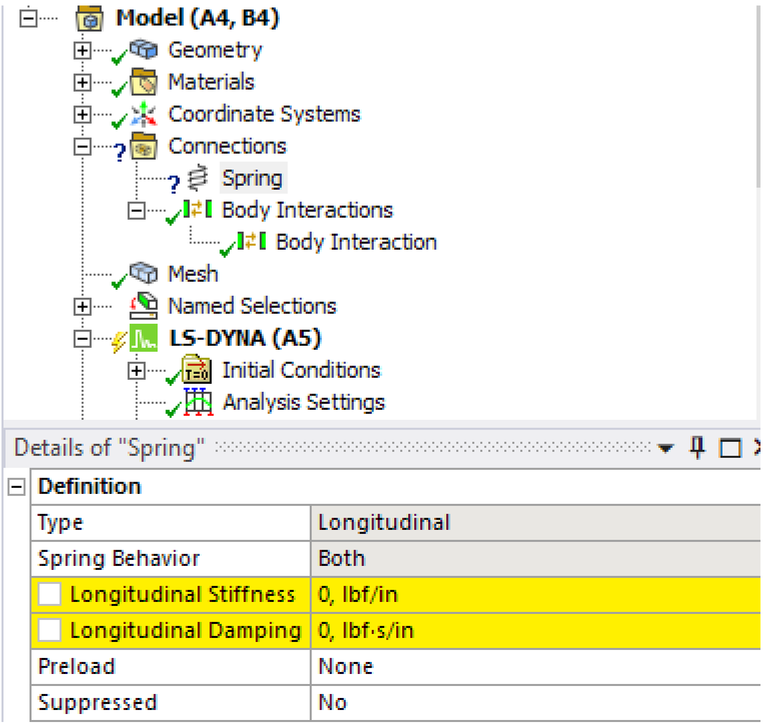
Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Type	Solid
ISPH Solid Material Controls	
Density	1.1 kg/m^3
Numerical Surface Adhesion	1.2 m/s^2
Surface Roughness Coefficient	1.3
SPH Mesh Surface Controls	
Space Between SPH Elements	0.1 m
Output SPH Mesh	No

3.1.4. Defining Connections

You can find information about Connections in Define Connections.

3.1.4.1. Springs

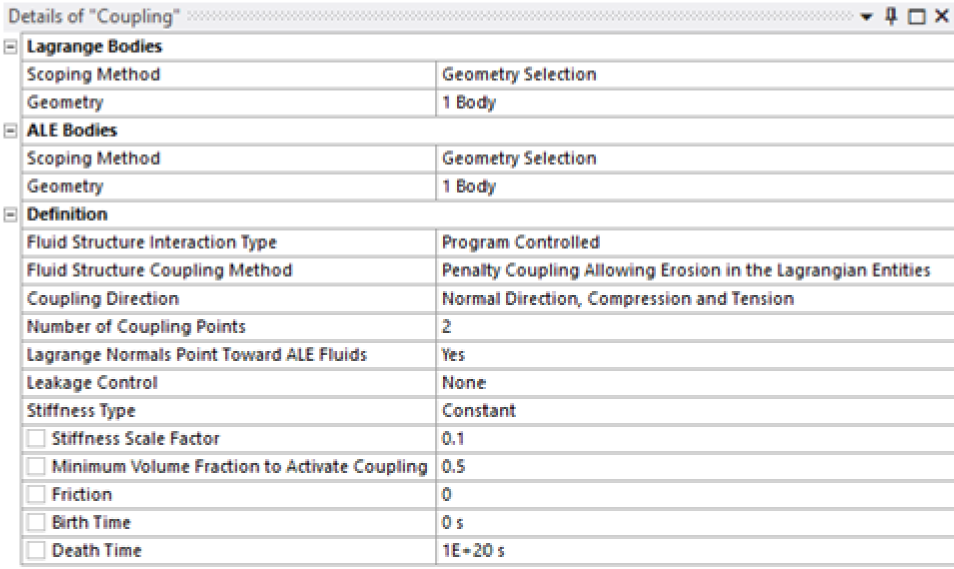
In LS-DYNA, Springs can be added in Connections. Both Longitudinal Stiffness and Damping are supported. For nonlinear springs, if you use Tabular Data to define the spring load curve, the force vs displacement curve must be defined with positive values for **Spring Behavior** set to Compression or to Tension. Negative values are valid only if the **Spring Behavior** is set to Both.



3.1.4.2. Coupling

A **Coupling** object can be added under **Connections** to define the coupling mechanism used when modeling Fluid-Structure Interaction (FSI). The coupling object determines the interaction between the scoped Lagrange Bodies (those with **Reference Frame** set to Lagrange) and ALE bodies (those with **Reference Frame** set to S-ALE Domain or S-ALE Fill, or contained in a section definition with an ALE element formulation). The **Coupling** object creates an instance of either the *CONSTRAINED_LAGRANGE_IN_SOLID (p. 161) or the *ALE_STRUCTURED_FSI keyword in the solver input file. The **Fluid Structure Interaction Type** field allows the keyword to be selected. The default Program Controlled option is *CONSTRAINED_LAGRANGE_IN_SOLID. The *ALE_STRUCTURED_FSI option is only applicable to S-ALE. Further information on the available options is given in the LS-DYNA Keyword User's Manual Volume I.

The Stiffness Type field allows you to define the stiffness as Constant (Stiffness Scale Factor) or Tabular (Penetration vs. Coupling Pressure).



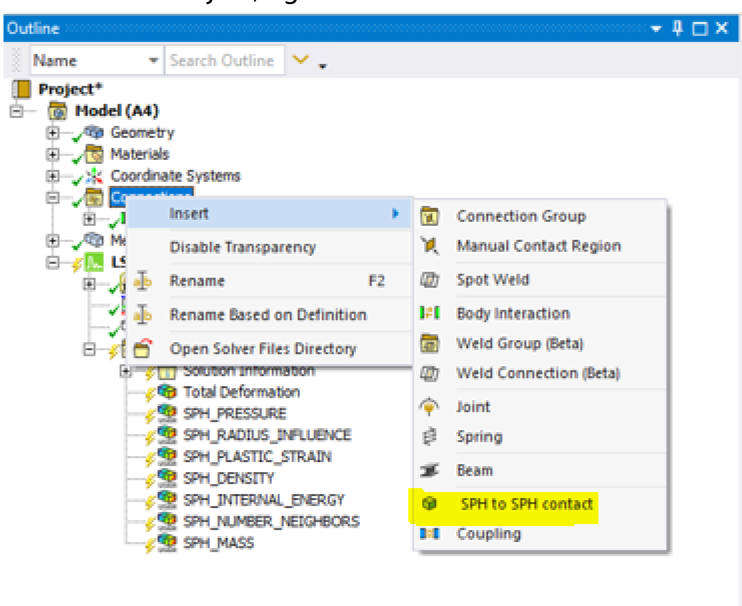
Details of "Coupling"

Lagrange Bodies	
Scoping Method	Geometry Selection
Geometry	1 Body
ALE Bodies	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Fluid Structure Interaction Type	Program Controlled
Fluid Structure Coupling Method	Penalty Coupling Allowing Erosion in the Lagrangian Entities
Coupling Direction	Normal Direction, Compression and Tension
Number of Coupling Points	2
Lagrange Normals Point Toward ALE Fluids	Yes
Leakage Control	None
Stiffness Type	Constant
Stiffness Scale Factor	0.1
Minimum Volume Fraction to Activate Coupling	0.5
Friction	0
Birth Time	0 s
Death Time	1E+20 s

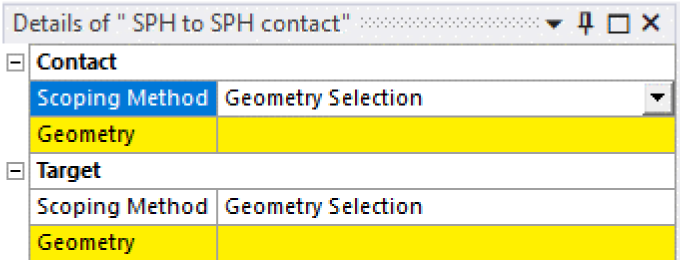
3.1.4.3. SPH to SPH Penalty-Based Contact

The SPH to SPH Contact object defines a penalty-based, node-to-node contact for particles of SPH parts. You can scope two SPH bodies that are interacting, and the solver will compute the interface stiffness value that guaranties stability (the interface stiffness is based on the nodal mass and the global time step size). This feature uses the keyword *DEFINE_SPH_TO_SPH_COUPLING and sets the parameter ISOFT to 1 (Soft constraint formulation).

To insert this object, right-click **Connections** and insert SPH to SPH Contact as shown here.



After the object is inserted, use the Details panel to scope a Contact body and a Target body.

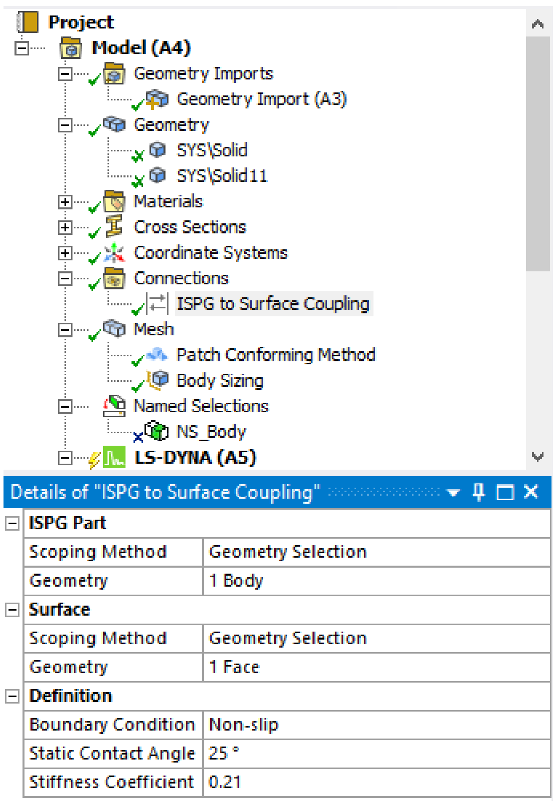


Scoping for this object is limited to SPH bodies and Adaptive Solid to SPH bodies. A warning message will be issued if incorrect scoping is attempted.

3.1.4.4. ISPG to Surface Coupling

The ISPG to Surface Coupling object defines a tied coupling interface between fluid particles modeled with implicit smoothed particle Galerkin (ISPG) and a surface. The part selected must be ISPG or an error will occur. This feature writes the keyword *DEFINE_FP_TO_SURFACE_COUPLING.

To insert this object, right-click **Connections** and select **Insert** → **ISPG to Surface Coupling**.

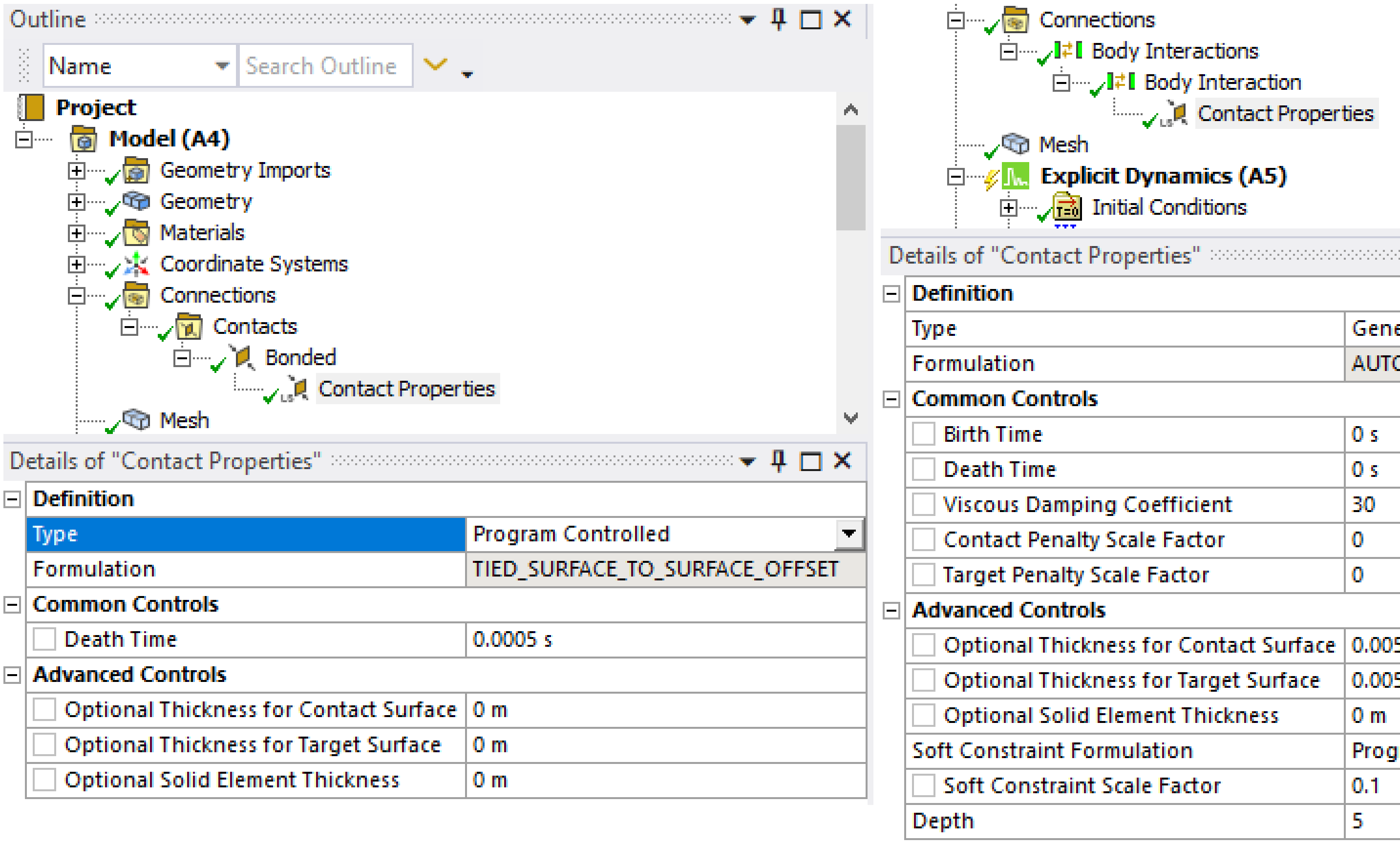


The **Static Contact Angle** is the angle between the surface and a tangent to the ISPG body passing through the contact point between the surface and ISPG body.

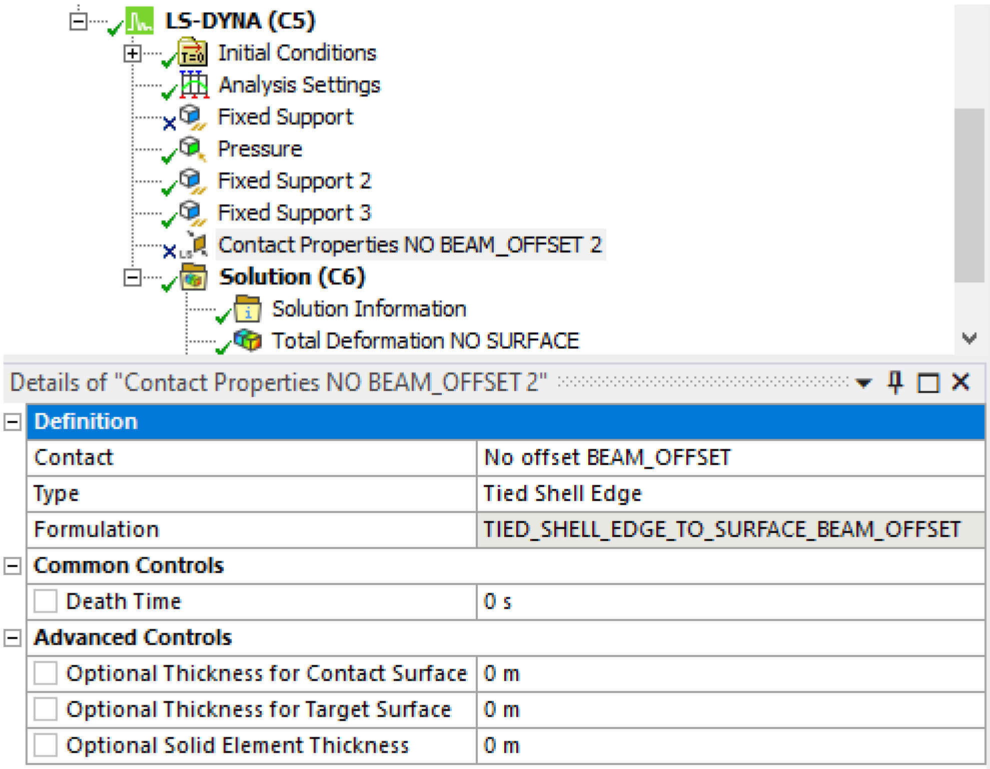
3.1.4.5. Additional Contact Properties

The Contact Properties object allows you to define LS-DYNA-specific contact options for a given Contact Region or a Body Interaction object. The object under analysis supersedes the object under Contact Region or Body Interaction if the two objects are scoped to the same contact.

Insert a Contact Properties object by right-clicking the contact you want to augment with LS-DYNA-specific properties, or by clicking the Contact Property icon on the **LSDYNA Pre** Tab.



You can also insert a Contact Properties object directly under an analysis environment. (You may want to do this in cases where you would like to have different properties for the same object in subsequent analyses). In this case, you can specify which contact you want the properties to apply to.



Note:

You can right-click a Contact Properties object that is under an analysis environment and select **Promote under Contact** to move it to an existing contact or body interaction.

For more information on the options available through the Contact Properties object see Keywords Created from the Contact Properties Object (p. 201).

3.1.5. Defining Mesh Settings

You can find information about Mesh Settings in Apply Mesh Controls/Preview Mesh.

Dimensionally Reduced Rigid Body Behavior is not available for LS-DYNA.

3.1.5.1. S-ALE Mesh

The mesh for bodies with the Reference Frame S-ALE Domain is created within the solver. There are two methods that can be used to define the mesh parameters that are passed to the solver. The default method takes the number of mesh elements created by the Mechanical meshing process and calculates the number of divisions along each coordinate axis needed to obtain a uniform rectilinear mesh.

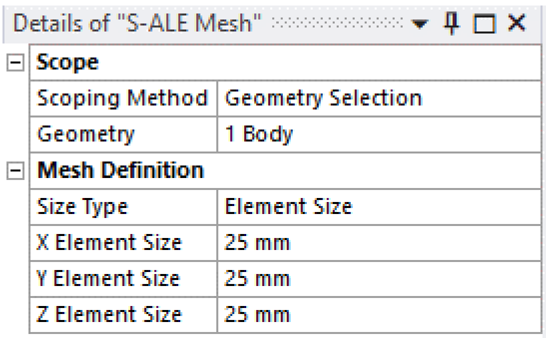
$$nx=\frac{dx}{cell\ size},\quad ny=\frac{dy}{cell\ size},\quad nz=\frac{dz}{cell\ size}$$

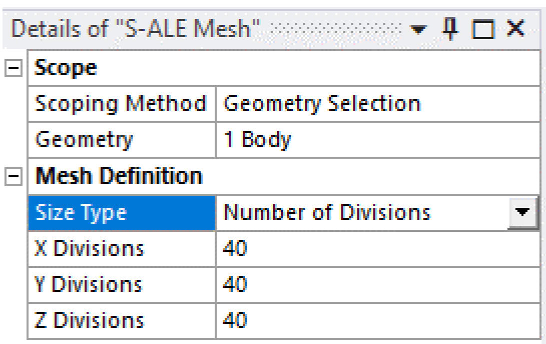
$$cell\ size=\sqrt[3]{\frac{dx*dy*dz}{N}}$$

Where:

- *N* = Number of elements in the Mechanical mesh
- *(dx , dy, dz)* = Length of the bounding box of the body along each of the coordinate axes
- *(nx, ny, nz)* = The number of divisions along each of the co-ordinate axes

The second method is to insert an **S-ALE Mesh** object under the **Model** branch. This object can be scoped to the S-ALE Domain body, and either the element size or the number of divisions along each co-ordinate axis can be defined directly. The object also provides a visual representation of the mesh on the exterior surface of the body.





3.1.6. Defining Named Selections

You can find information about creating a **Named Selection** in Named Selections in the *Mechanical User's Guide*.

The **LS-DYNA Named Selection User ID** field is only visible when a **Named Selection** object is created while performing an analysis using an LS-DYNA system in Workbench. This field allows you to define an id that will be used by the LS-DYNA solver for the named selection that you created. The value of the field can be:

- 0 - The id will be program controlled
- Positive Integer - The number entered will be used by LS-DYNA as the id for the named selection

- **Note:**

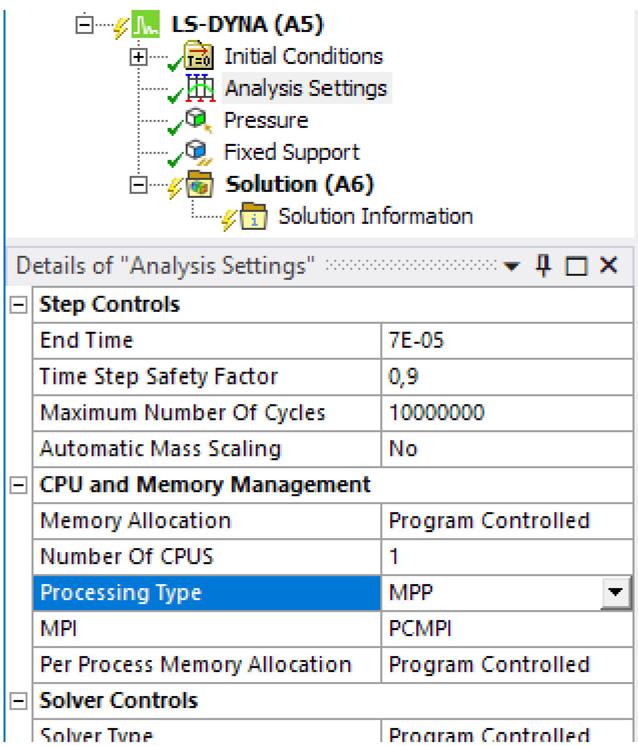
  If you read in an LS-DYNA input file that contains named selections, the **LS-DYNA Named Selection User ID** will not be set to the corresponding value for each **Named Selection** object created in the project. However, you can set the id of each named selection by hand if you change the **Transfer PropertiesRead Only** field in the Details panel of the Named Selection to No.

3.1.7. Defining Analysis Settings

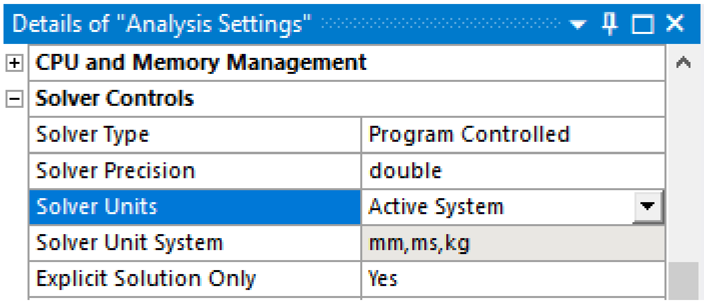
You can find information about **Analysis Settings** in Establish Analysis Settings.

You can set **Solver Precision** to Single or Double under **Solver Controls**, or leave it as Program Controlled.

In LS-DYNA, the **End Time** is the only required input. SMP or MPP parallel processing can be activated by changing the Number of CPU's. Note that Ansys LS-DYNA HPC licenses are required for using more than 1 core.

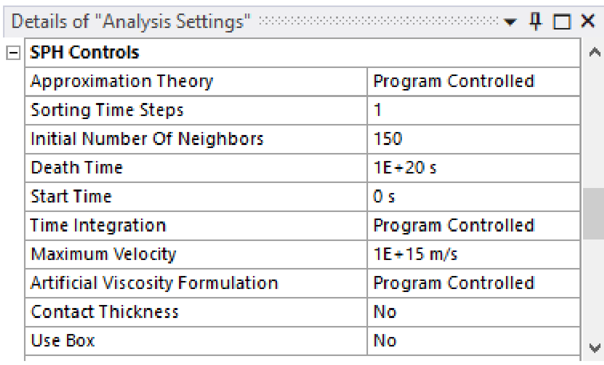


The property **Solver Units** in the Solver Controls category has two options, Active System or Manual. If set to Active System the **Solver Unit System** property (which sets the solver unit system) will be read only and its value will determined by the GUI Unit System (set via the **Units** option in the **Tools** group of the Home tab on the ribbon). If set to Manual, the solver unit system can be set using the **Unit System** property and can be different from the GUI Unit System. This follows the standard Mechanical behavior described here.



SPH Controls

For the SPH solver, the following settings are available in the SPH Controls section of the Details panel in LS-DYNA. For more information, see *CONTROL_SPH (p. 139).



Field	Description
Approximation Theory	Particle approximation method: • Program Controlled • Default Formulation • Renormalization Approximation • Fluid Particle Approximation • Fluid Particle With Renormalization Approximation • Total Lagrangian Formulation • Total Lagrangian Formulation With Renormalization • Enhanced Fluid Formulation • Enhanced Fluid Formulation With Renormalization • Moving Least Squares (can only be used with a **Processing Type** of MPP) • ISPH (can only be used with a **Processing Type** of MPP)
Sorting Time Steps	Number of time steps between particle sorting

Field	Description
Initial Number of Neighbors	Initial number of neighbors per particle
Death Time	Death time for particle approximation
Start Time	Start time for particle approximation
Time Integration	Time integration type for the smoothing length: • Program Controlled • Default • Enhanced Energy Formulation
Maximum Velocity	Maximum velocity For SPH particles
Artificial Viscosity Formulation	Artificial viscosity formulation for SPH elements: • Program Controlled • Monaghan Formulation • Standard Formulation
Contact Thickness	If Yes, the solver will define a contact thickness in order to detect and avoid penetration between an SPH part and a Lagrangian part
Use Box	If Yes, exposes a **Box Name** field that lets you specify a standard boundary box (p. 60) to be used during the analysis

ALE Controls

For the ALE solver, the following settings are available in the ALE Controls section of the Details panel in LS-DYNA. For more information, see *CONTROL_ALE (p. 131).

ALE Controls	
Continuum Treatment	Use Alternate Advection Logic
Cycles Between Advection	1
Advection Method	Donor Cell + Half Index Shift
Simple Average Weighting Factor	-1
Volume Weighting Factor	0
Isoparametric Weighting Factor	0
Equipotential Weighting Factor	0
Equilibrium Weighting Factor	0
Advection Factor	0
Start	0 s
End	1E+20 s
Reference Pressure	0 Pa
Pressure Equilibrium Iteration	No

Field	Description
Continuum Treatment	Flag to invoke alternate advection logic: • Use Alternate Advection Logic • Use Default Advection Logic
Cycles Between Advection	Number of cycles between advections (almost always set to 1).
Advection Method	Advection methods: • Donor cell + Half-Index-Shift • Van Leer + Half-Index-Shift • Van Leer (Relaxed Monotonicity) • Donor Cell (Total Energy Conservation) • Finite Volume With Flux Corrected Transport
Simple Average Weighting Factor	ALE smoothing weight factor - Simple average
Volume Weighting Factor	ALE smoothing weight factor - Volume weighting
Isoparametric Weighting Factor	ALE smoothing weight factor - Isoparametric
Equipotential Weighting Factor	ALE smoothing weight factor - Equipotential
Equilibrium Weighting Factor	ALE smoothing weight factor - Equilibrium
Advection Factor	ALE advection factor (donor cell options, default = 1.0). This field is obsolete.
Start	Start time for ALE smoothing or start time for ALE advection if smoothing is not used.
End	End time for ALE smoothing or end time for ALE advection if smoothing is not used.
Reference Pressure	Applied to the free surfaces of the ALE domain.
Pressure Equilibrium Iteration	A flag to turn on or off the pressure equilibrium iteration option for multi-material elements.

3.2. Using Symmetry

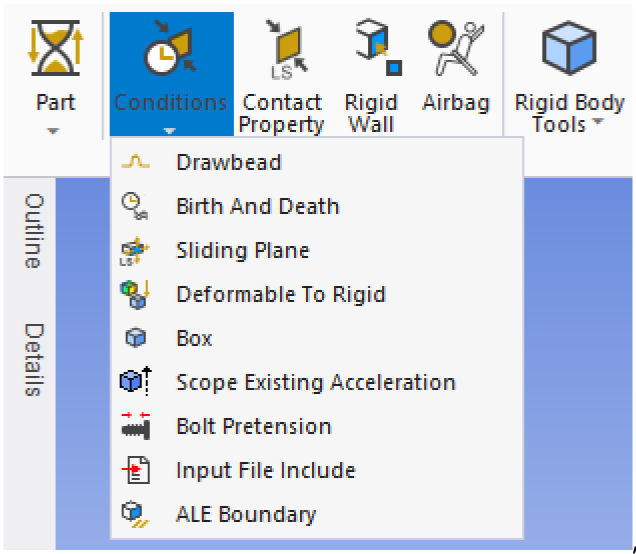
Cyclic Symmetry is supported in LS-DYNA analyses. For more information on setting up Cyclic Symmetry, see Symmetry.

3.3. Defining Initial Conditions

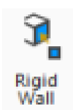
You can find information about Initial Conditions in Define Initial Conditions.

3.4. Defining Boundary Conditions

You can find information about Boundary Conditions in Apply Loads and Supports. For an LS-DYNA system, Remote Displacement is available in addition to the boundary conditions discussed there.

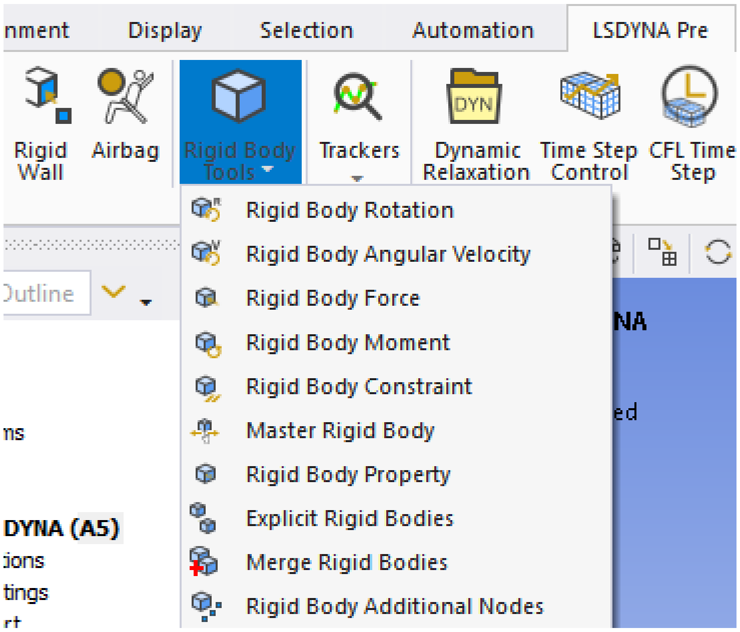
The Conditions 

, Contact Property 

, and Rigid Wall (p. 216) 

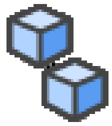
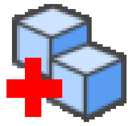
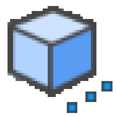
constraints are available on the LSDYNA Pre tab. The following boundary conditions are also available:

3.4.1. Rigid Body Tools



Use the **Rigid Body Tools** to specify the behavior of the rigid bodies in your model.

- Rigid Body Rotation (p. 40)  - specify the rotation of a rigid body(ies).
- Rigid Body Angular Velocity (p. 42)  - specify the angular velocity of a rigid body(ies).
- Rigid Body Force (p. 44)  - apply a force to the center of mass of a rigid body(ies).
- Rigid Body Moment (p. 46)  - apply a moment to a rigid body(ies).
- Rigid Body Constraint (p. 47)  - constrain the motion of a rigid bod(dies).
- Master Rigid Body (p. 48)  - link multiple rigid bodies together. The master rigid body leads the movement of the other linked bodies.
- Rigid Body Property (p. 49)  - set the mass, center of mass, and mass moment of inertia of a rigid body.

- Explicit Rigid Bodies (p. 50)  - identify body(ies) that are rigid in an LS-Dyna Explicit analysis and flexible in other types of analyses.
- Merge Rigid Bodies (p. 50)  - merge multiple rigid bodies together into a single rigid body.
- Rigid Body Additional Nodes (p. 51)  - add additional nodes from a flexible body to an existing rigid body.

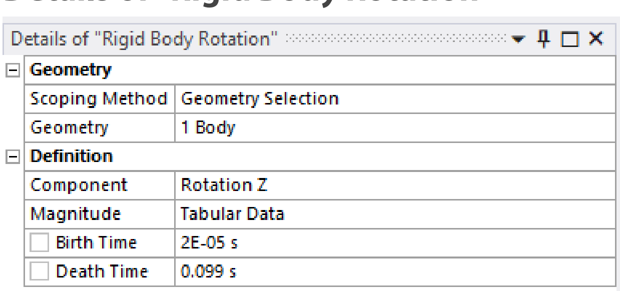
3.4.1.1. Rigid Body Rotation

Use the **Rigid Body Rotation** tool  to specify the rotation of a rigid body (or bodies) in radians. This command rotates the selected rigid body(ies) around the x-axis, y-axis, or z-axis of the global coordinate system according to the rotation curve you enter.

Select the rigid body(ies) to be rotated, specify the axis of rotation, and define the times and magnitudes of the rotation curve in a table. For more complex rotations, break the rotation down into its x-, y-, and z-components and define separate rigid body rotation objects for rotation around each axis.

During the solution, the LS-DYNA solver applies a force to rotate the rigid body(ies) according to the rigid body rotation direction(s) and curve(s) you entered. It starts each rotation curve at the birth time and stops it at the death time.

Details of "Rigid Body Rotation"



Details of "Rigid Body Rotation"

Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Component	Rotation Z
Magnitude	Tabular Data
Birth Time	2E-05 s
Death Time	0.099 s

Specify the input parameters for the **Rigid Body Rotation** tool:

Field	Description
Scoping Meth-od	Choose the method for selecting one or more rigid bodies. Only select objects that were defined as rigid bodies. • **Geometry Selection** - click a rigid body to select it. • **Named Selection** - choose the name of a rigid body from the drop-down list.
Geometry	Displays the number of selected rigid bodies.
Component	Choose the axis of rotation for the selected rigid body(ies): **Rotation X**, **Rotation Y**, or **Rotation Z**. Rotation can be specified independently for each axis.
Magnitude	Enter the time steps and magnitude of the rotation of the selected rigid bodiy(ies) (that is, the rotation curve). See Tabular Data for Rigid Body Rotation (p. 41) (below).
Birth Time	Enter the time when the LS-DYNA solver begins executing the rotation curve defined under Tabular Data for Rigid Body Rotation (p. 41). The boundary condition becomes active at that time and is used in the solution. The **Birth Time** is set independently of the rotation start time defined under **Tabular Data**.
Death Time	Enter the time when the LS-DYNA solver stops executing the rotation curve defined under Tabular Data for Rigid Body Rotation (p. 41). The boundary condition becomes inactive and is no longer used in the solution. The **Death Time** is set independently of the rotation end time defined under **Tabular Data**.

Tabular Data for Rigid Body Rotation

Tabular Data

		Time [s]	✔ Magnitude [rad]
1	1	0.	0.
2	1	2.e-002	0.7854
3	1	8.e-002	0.
4	1	8.0001e-002	1.5708
5	1	0.1	1.5708
*			

Use this table to enter a rotation curve that specifies how the selected rigid body(ies) rotate. At each step in the curve, enter the time and magnitude of rotation.

Column	Description
1, **2**, **3** … *	The first column shows the time step of the rotation curve. Each time step is numbered in order of its execution during the solution. Enter a new time and magnitude into the * field. See Adding Time Steps to the Rigid Body Rotation Curve (p. 42) (below).
1, **N/A**	The second column shows whether the time step occurs (and is used) during the solution • A number indicates rotation time steps that occur during the solution.

Column	Description
	• **N/A** indicates time steps that occur after the solution ends.
Time [s]	The time in seconds at which rotation occurs. The first time step defaults to zero. The last time step defaults to the end of the solution time.
Magnitude [rad]	The magnitude of rotation in radians.

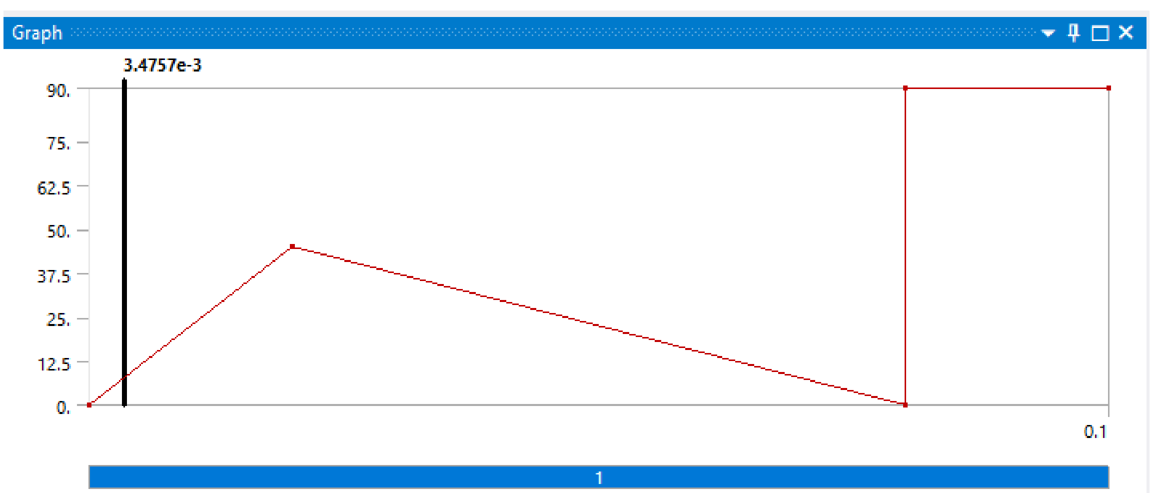
Adding Time Steps to the Rigid Body Rotation Curve

You can manually enter a **Time** and **Magnitude** of rotation in the row marked with *. Alternatively, you can copy and paste pairs of time and magnitude values from a spreadsheet into that row.

The system adds these values to the table according to their **Time** value.

- If you enter a **Time** between the start time (time step 1) and the end time (the last numbered time step in the table), the system adds the time step to the table in order of its time value.
- If you enter a **Time** without a corresponding **Magnitude**, the system automatically assigns a rotation value that is interpolated between the two surrounding time steps. You can then change the magnitude of rotation to the desired value.
- If you enter a time value after the end time, the system adds it to the table as an **N/A** row. It is not used during the solution.

Graph of Rigid Body Rotation Curve



The **Graph** panel plots the rotation curve that you defined under **Tabular Data**. Click the check box next to **Magnitude [rad]** in the **Tabular Data** table to toggle the display of this curve.

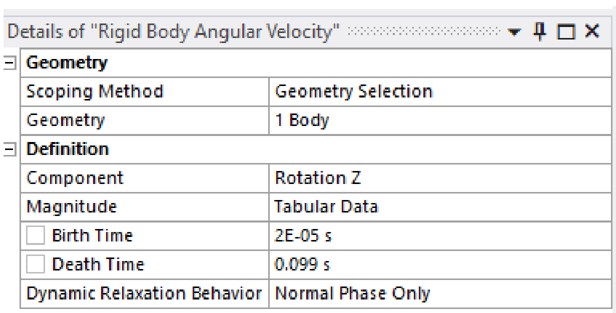
3.4.1.2. Rigid Body Angular Velocity

Use the **Rigid Body Angular Velocity** tool  to set the angular velocity of the rigid body in radians per second . This command applies an angular velocity to the selected rigid body(ies) around the x-axis, y-axis, or z-axis of the global coordinate system according to the angular velocity curve you specify.

Select the rigid body(ies) to which an angular velocity is applied, specify the axis of rotation, and define the angular velocity curve (that is, the times and magnitudes of angular velocity) in a table. For more complex rotation patterns, break the angular velocity down into its x-, y-, and z-components and define separate rigid body angular velocity objects for the angular velocity around each axis.

During the solution, the LS-DYNA solver applies a force to the rigid body to rotate it according to the angular velocity direction(s) and curve(s) you entered. It starts to execute each angular velocity curve at the birth time and stops it at the death time.

Details of "Rigid Body Angular Velocity"


Details of "Rigid Body Angular Velocity"

Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Component	Rotation Z
Magnitude	Tabular Data
Birth Time	2E-05 s
Death Time	0.099 s
Dynamic Relaxation Behavior	Normal Phase Only

Specify the input parameters for the **Rigid Body Angular Velocity** tool:

Field	Description
Scoping Method	Choose the method for selecting one or more rigid bodies. See the discussion of **Scoping Method** under Details of "Rigid Body Rotation" (p. 40).
Geometry	Displays the number of selected rigid bodies.
Component	Select the axis of rotation for the selected rigid body(ies). See the discussion of **Component** under Details of "Rigid Body Rotation" (p. 40).
Magnitude	Enter the time steps and magnitude of the angular velocity of the selected rigid body(ies) (that is, the angular velocity curve). See Tabular Data for Rigid Body Angular Velocity (p. 44) (below).
Birth Time	Enter the time when the LS-DYNA solver begins executing the angular velocity curve that is defined under Tabular Data for Rigid Body Angular Velocity (p. 44). The boundary condition becomes active at that time and is used in the solution. The **Birth Time** is set independently of the angular velocity start time defined under **Tabular Data**.
Death Time	Enter the time that the solver stops executing the rigid body angular velocity curve that is defined under Tabular Data for Rigid Body Angular Velocity (p. 44). The boundary condition becomes inactive and is no longer used in the solution. The **Death Time** is set independently of the angular velocity end time defined under **Tabular Data**.

Field	Description
Dynamic Relaxation Behavior	Dynamic relaxation sets the preloading for explicit dynamics solutions in LS-DYNA. Select one of the following: • **Normal Phase Only** • **Dynamic Relaxation Phase Only** • **Both** See Dynamic Relaxation (p. 56) for more information.

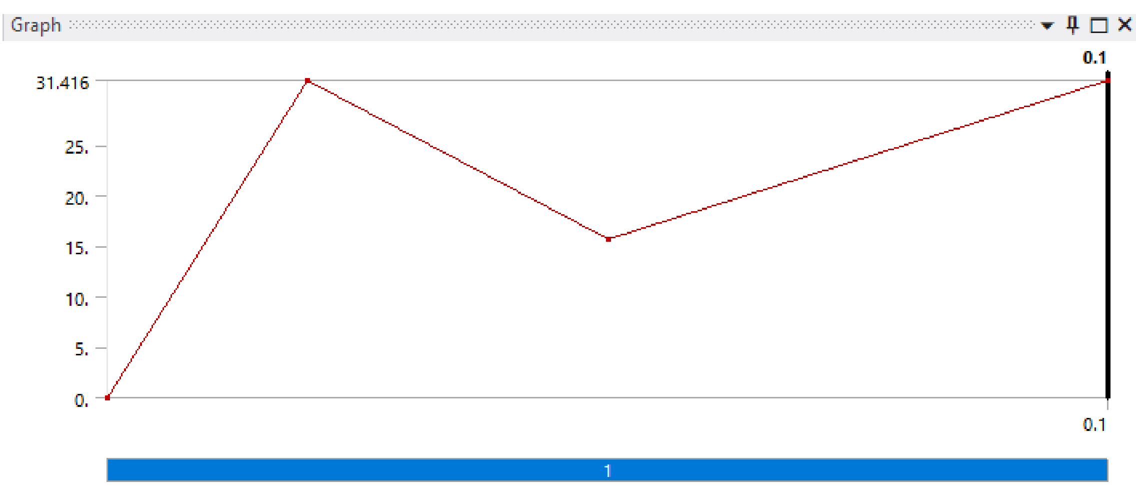
Tabular Data for Rigid Body Angular Velocity

Tabular Data

		Time [s]	Magnitude [rad/s]
1	1	0.	0.
2	1	2.0001e-002	31.416
3	1	5.e-002	15.708
4	1	0.1	31.416
*			

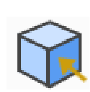
Use this table to enter the angular velocity curve, which specifies how the system applies angular velocities to the selected rigid body(ies). Enter the **Time** in seconds and the **Magnitude** of the angular velocity in radians per second. See Tabular Data for Rigid Body Rotation (p. 41) for more information.

Graph of Rigid Body Angular Velocity Curve



The **Graph** panel plots the angular velocity curve that you defined under **Tabular Data**. Click the check box next to **Magnitude [rad/s]** in the **Tabular Data** table to toggle the display of this curve.

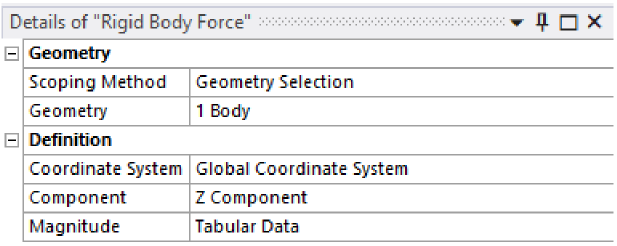
3.4.1.3. Rigid Body Force

Use the **Rigid Body Force** tool  to apply a force to a rigid body. This command applies a force in the x, y, or z direction to the center of mass of the selected rigid body(ies), according to the force curve you specify.

Select the rigid body(ies) to which a force is applied, specify the direction, and define the force curve (that is, the times and magnitudes at which force is applied) in a table. You can decompose a force vector into its x, y, and z components and define separate rigid body force objects for each component.

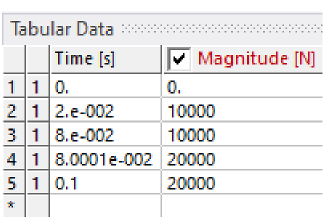
During the solution, the LS-DYNA solver applies a force to the rigid body according to the force direction(s) and curve(s) you entered. The solver starts to execute each force curve at the start time under **Tabular Data** and stops it at the end time.

Details of "Rigid Body Force"



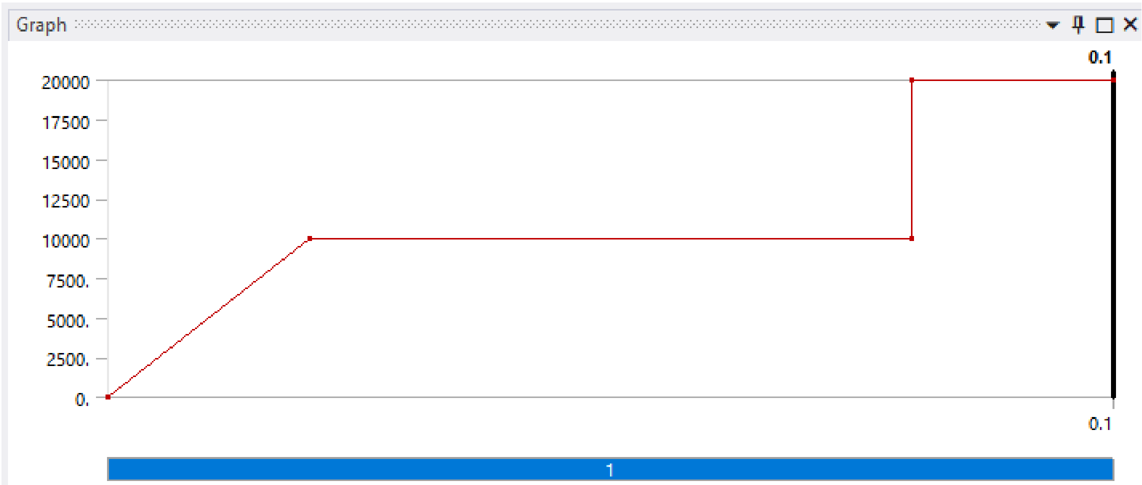
Field	Description
Scoping Method	Choose the method for selecting one or more rigid bodies. See the discussion of **Scoping Method** under Details of "Rigid Body Rotation" (p. 40).
Geometry	Displays the number of selected bodies.
Coordinate System	Specifies the coordinate system (defaults to **Global Coordinate System**)
Component	Specify which component of the force vector you are defining (**X Component**, **Y Component**, or **Z Component**).
Magnitude	Enter the time steps and magnitude of the force that is applied to the selected rigid body(ies) (that is, the force curve). See Tabular Data for Rigid Body Force (p. 45) (below).

Tabular Data for Rigid Body Force



Use this table to enter the force curve, which specifies how the system applies a force to the selected rigid body(ies). Enter the **Time** in seconds and the **Magnitude** of the force in Newtons. See Tabular Data for Rigid Body Rotation (p. 41) for more information.

Graph of Rigid Body Force Curve



The **Graph** panel plots the force curve that you defined under **Tabular Data**. Click the check box next to **Magnitude [N]** in the **Tabular Data** table to toggle the display of this curve.

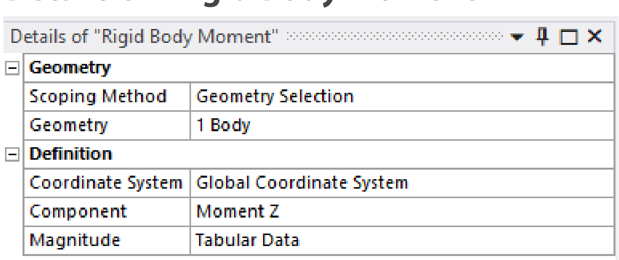
3.4.1.4. Rigid Body Moment

Use the **Rigid Body Moment** tool  to apply a moment to a rigid body. This command applies a moment in the x, y, or z direction to the selected rigid body(ies), according to the moment curve you specify.

Select the rigid body(ies) to which a moment is applied, specify the direction, and define the moment curve (that is, the times and magnitudes at which a moment is applied) in a table. You can decompose a moment vector into its x, y, and z components and define separate rigid body moment objects for each component.

During the solution, the LS-DYNA solver applies a moment to the rigid body according to the direction(s) and curve(s) you entered. The solver starts to execute each moment curve at the start time under **Tabular Data** and stops it at the end time.

Details of "Rigid Body Moment"

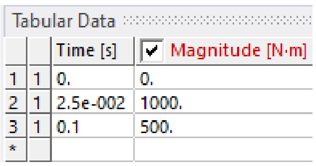

Details of "Rigid Body Moment"

Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Coordinate System	Global Coordinate System
Component	Moment Z
Magnitude	Tabular Data

Field	**Description**
Scoping Method	Choose the method for selecting one or more rigid bodies. See the discussion of **Scoping Method** under Details of "Rigid Body Rotation" (p. 40).
Geometry	Lists the number of selected bodies.

Field	Description
Coordinate System	Specifies the coordinate system (defaults to **Global Coordinate System**)
Component	Specify which component of the moment vector you are defining (**Moment X** , **Moment Y** , or **Moment Z**).
Magnitude	Enter the time steps and magnitude of the moment that is applied to the selected rigid body(ies) (that is, the force curve). See Tabular Data for Rigid Body Moment (p. 47) (below).

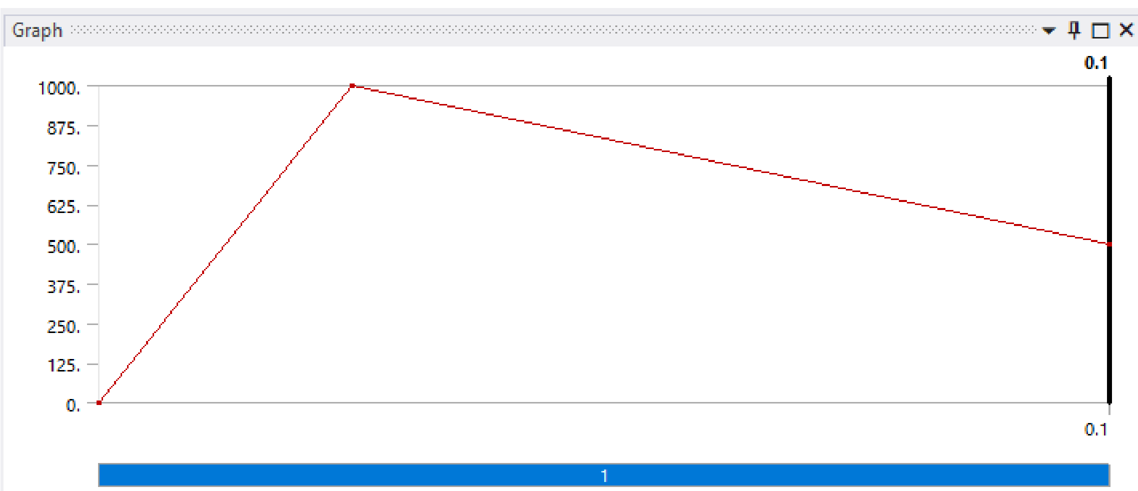
Tabular Data for Rigid Body Moment


Tabular Data

		Time [s]	Magnitude [N·m]
1	1	0.	0.
2	1	2.5e-002	1000.
3	1	0.1	500.
*			

Use this table to enter the moment curve, which specifies how the system applies a moment to the selected rigid body(ies). Enter the **Time** in seconds and the **Magnitude** of the force in Newton-meters. See Tabular Data for Rigid Body Rotation (p. 41) for more information.

Graph of Rigid Body Moment Curve



The **Graph** panel plots the moment curve that you defined under **Tabular Data**. Click the check box next to **Magnitude [N·m]** in the **Tabular Data** table to toggle the display of this curve.

3.4.1.5. Rigid Body Constraint

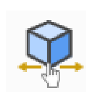
Use the **Rigid Body Constraint** tool  to constrain the motion of a rigid body. This tool allows you to specify the degrees of freedom for the selected rigid body(ies). You can constrain their movement in the x, y, or z direction and their rotation around the global x, y, or z axis.

Details of "Rigid Body Constraint"

Details of "Rigid Body Constraint"	
Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
X Component	Fixed
Y Component	Fixed
Z Component	Fixed
Rotation X	Fixed
Rotation Y	Free
Rotation Z	Fixed

Field	Description
Scoping Meth-od	Choose the method for selecting one or more rigid bodies. See the discussion of **Scoping Method** under Details of "Rigid Body Rotation" (p. 40).
Geometry	Lists the number of selected bodies.
X Component **Y Compon-ent** **Z Component**	Choose whether the rigid body's movement in the x, y or z direction is **Fixed** (cannot move in that direction) or **Free** (no restriction on movement in that direction).
Rotation X **Rotation Y** **Rotation Z**	Choose whether the rigid body's rotation around the x, y or z axis is **Fixed** (cannot rotate around that axis) or **Free** (no restriction on rotation around that axis).

3.4.1.6. Master Rigid Body

Use the **Master Rigid Body** tool  to link two or more rigid bodies together. The rigid bodies must be connected by a mesh.

In the Mechanical workbench, the components of a larger body can be grouped together. When this group contains rigid bodies, the LS-DYNA solver merges the rigid bodies and computes a single solution for them. By using the **Master Rigid Body** tool, you can pick which rigid body in this group is assigned boundary conditions (the master rigid body). The solver applies any rigid body boundary conditions to the master rigid body you select. The master rigid body leads the movement of the rigid bodies that are connected to it by a mesh. The connected rigid bodies are constrained to move with the master rigid body.

Use the Merge Rigid Bodies (p. 50) tool to combine rigid bodies that are not connected by a mesh.

Details of "Master Rigid Body"

Details of "Master Rigid Body"	
Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Body	Solid

Select the master rigid body.

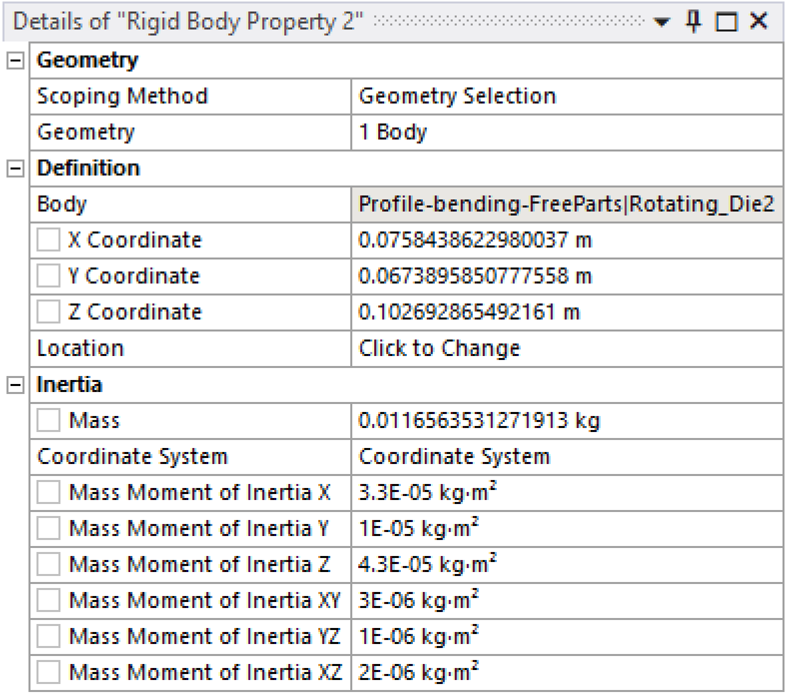
Field	Description
Scoping Method	Specify the method for selecting a body. See the discussion of **Scoping Method** under Details of "Rigid Body Rotation" (p. 40).
Geometry	Lists the number of selected bodies.
Body	The type of body selected as the master rigid body.

3.4.1.7. Rigid Body Property

Use the **Rigid Body Property** tool  to set the following rigid body properties:

Field	Description
Scoping Method	Specify the method for selecting a rigid body. Only select objects that were defined as rigid bodies. • **Geometry Selection** - click a rigid body to select it. The **Geometry** field shows the number of selected bodies. • **Named Selection** - choose the name of a rigid body from the drop-down list.
Body, Coordinates	After the Geometry is scoped to the Rigid Body Property object, the name of the body is displayed and the coordinates of the center of mass are transferred from the scoped body to the Coordinates fields.
Location	Use this field to change the location of the center of mass of the body.
Mass, Moments of Inertia	The Mass and Moments of Inertia are transferred from the scoped body to these fields.
Coordinate System	Select the coordinate system in which you want to display the center of mass.

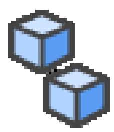
The Rigid Body Property tool allows you to set the mass, center of mass, and mass moment of inertia of a rigid body. When scoping a body to the geometry, the name of the Body is displayed, and the properties X,Y,Z coordinates, mass, and moments of inertia are filled accordingly from the values in the Geometry object. You can change the center of mass by scoping a geometric entity to the Location. When scoping multiple geometric entities, the equivalent center of mass is computed. The equivalent center of mass is the weighted mean of the center of gravity times the mass of all geometric entities. If you change the mass, the mass moments of inertia will be multiplied by the ratio between the new entered mass by the already scoped bodies mass.



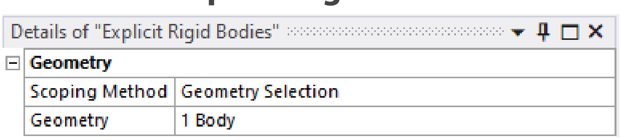
Details of "Rigid Body Property 2"

Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Body	Profile-bending-FreeParts|Rotating_Die2
X Coordinate	0.0758438622980037 m
Y Coordinate	0.0673895850777558 m
Z Coordinate	0.102692865492161 m
Location	Click to Change
Inertia	
Mass	0.0116563531271913 kg
Coordinate System	Coordinate System
Mass Moment of Inertia X	3.3E-05 kg·m²
Mass Moment of Inertia Y	1E-05 kg·m²
Mass Moment of Inertia Z	4.3E-05 kg·m²
Mass Moment of Inertia XY	3E-06 kg·m²
Mass Moment of Inertia YZ	1E-06 kg·m²
Mass Moment of Inertia XZ	2E-06 kg·m²

3.4.1.8. Explicit Rigid Bodies

Use the **Explicit Rigid Bodies** tool  to identify body(ies) that are treated as being rigid in an LS-DYNA Explicit analysis and flexible in other types of analyses.

Details of "Explicit Rigid Bodies"



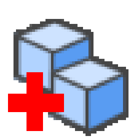
Details of "Explicit Rigid Bodies"

Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body

Select the body(ies) that are treated as rigid during an Explicit analysis.

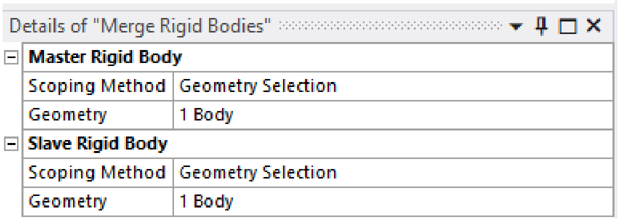
Field	Description
Scoping Method	Specify the method for selecting a body. See the discussion of **Scoping Method** under Details of "Rigid Body Rotation" (p. 40).
Geometry	Lists the number of selected bodies.

3.4.1.9. Merge Rigid Bodies

Use the **Merge Rigid Bodies** tool  to combine multiple rigid bodies together into a single rigid body. The rigid bodies do not have to be adjacent.

Ordinarily, the LS-DYNA solver considers each rigid body separately. By using the **Merge Rigid Body** tool, you can combine the selected rigid bodies into a single body. All rigid body boundary conditions are applied to the master rigid body. The slave rigid bodies are constrained to move with the master rigid body.

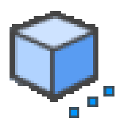
Details of "Merge Rigid Bodies"



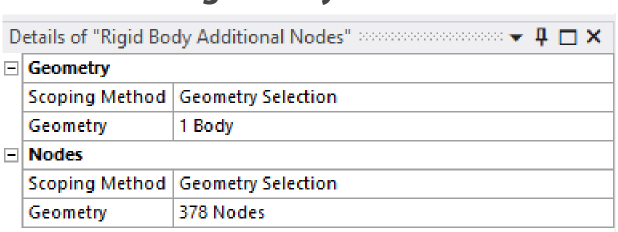
Select the **Master Rigid Body** and at least one **Slave Rigid Body**.

Field	Description
Scoping Method	Specify the method for selecting a rigid body. See the discussion of **Scoping Method** under Details of "Rigid Body Rotation" (p. 40).
Geometry	Lists the number of selected bodies.

3.4.1.10. Rigid Body Additional Nodes

Use the **Rigid Body Additional Nodes** tool  to create additional nodes in the mesh between adjacent rigid and flexile bodies. This adds rigid nodes to the flexible body, which creates a better connection and prevents deformation between the rigid and flexible bodies.

Details of "Rigid Body Additional Nodes"



Under **Geometry**, select the desired rigid body. Under **Nodes**, select the desired entities (non-body geometry scopings or nodes) to be added to the selected rigid body. The system adds nodes to the mesh for the selected entities.

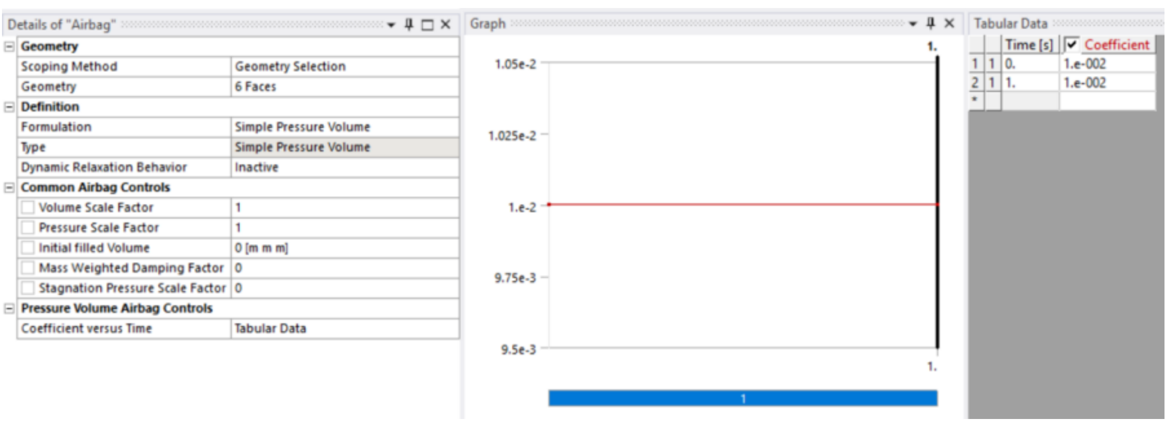
Field	Description
Scoping Method	Specify the method for selecting a body. See the discussion of **Scoping Method** under Details of "Rigid Body Rotation" (p. 40).
Geometry	**Geometry** - Lists the number of selected bodies. **Nodes** - Lists the number of nodes that are added.

3.4.2. Airbag or Simple Pressure Volume

The Airbag object provides a way of defining thermodynamic behavior of the gas flow into the airbag as well as a reference configuration for the fully inflated bag.

Details of "Airbag"	
Geometry	
Scoping Method	Geometry Selection
Geometry	6 Faces
Definition	
Formulation	Simple Pressure Volume
Type	Simple Pressure Volume
Common Airbag Controls	
Volume Scale Factor	1
Pressure Scale Factor	1
Initial filled Volume	0 [m m m]
Mass Weighted Damping Factor	500
Stagnation Pressure Scale Factor	0
Pressure Volume Airbag Controls	
Coefficient versus Time	Tabular Data

Simple Pressure Volume



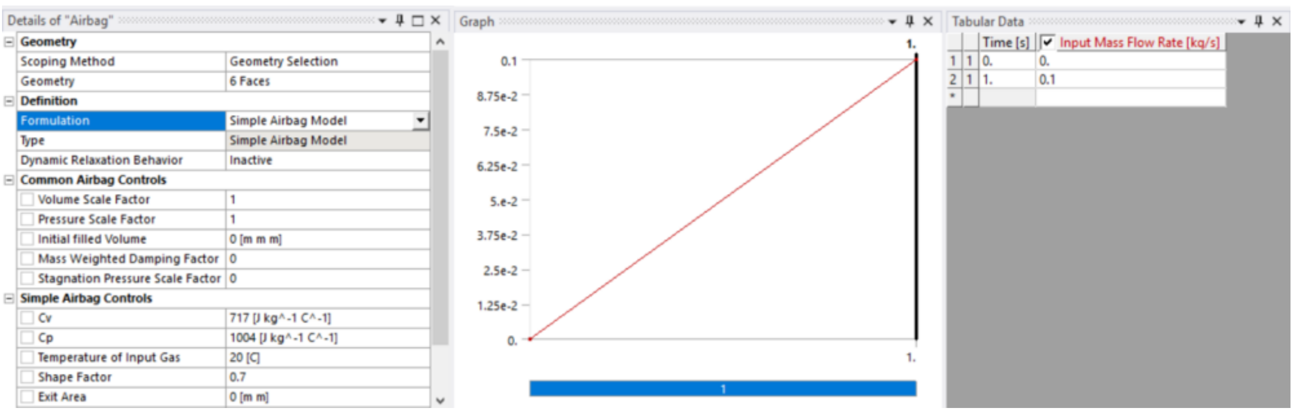
When **Formulation** is set to Simple Pressure Volume:

- The pressure is a function of the ratio of current volume to the initial volume.
- This simple model can be used when an initial pressure is given.
- No leakage, no temperature change, and no input mass flow are assumed.
- A typical application is the modeling of air in automobile tires.
- Pressure Volume Airbag controls: Coefficient versus Time (CN) can be defined in a table. β is a scale factor for the curve defining the coefficient versus time (CN), with a default value of 1.

$$Pressure = \beta \frac{CN}{Relative\ Volume}$$

$$Relative\ Volume = \frac{Current\ Volume}{Initial\ Volume}$$

Simple Airbag Model



The volume pressure relationships is defined by the Simple Airbag Model for control volumes.

- The gamma law equation of state used to determine the pressure in the airbag:

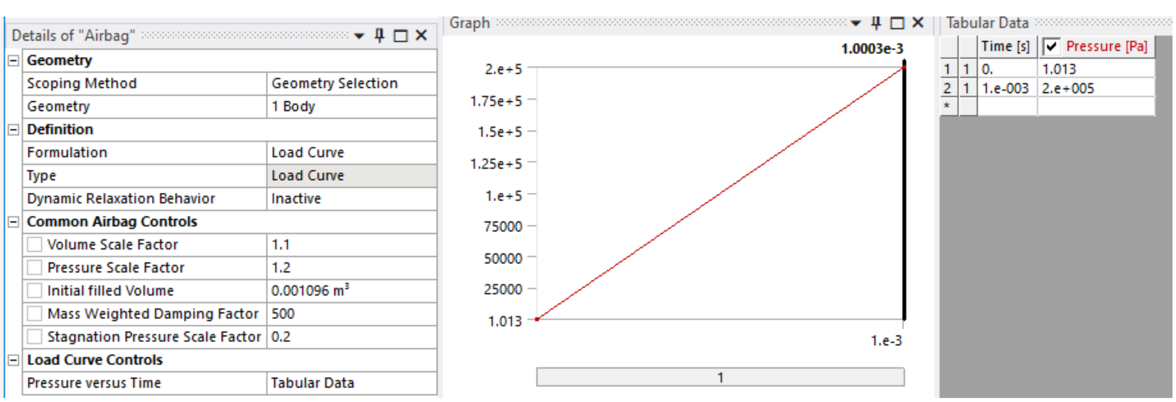
  $p = (\gamma\text{-}1)\rho e$

  where p is the pressure, ρ is the density, e is the specific internal energy of the gas, and γ is the ratio of the specific heats:

  $\gamma = c_p/c_v$

- Input Mass Flow Rate can be defined in a table.

Load Curve Airbag Model



You can supply a tabular load curve to define the pressure as a function of time.

3.4.3. Input File Include Constraint

Advanced users can insert additional LS-DYNA keywords by using the **Input File Include** constraint to specify the name of a file that contains LS-DYNA keywords. An include file `*INCLUDE Filename1` keyword card will be generated pointing to the file you specify.

Included files can contain any valid LS-DYNA keyword cards. Using include files eliminates the need for you to edit the `.k` input file each time the file is created by LS-DYNA if you have other keywords you want to use. Note that the included file can contain other include file statements, providing a fairly general capability for easily adding predefined inputs.

3.4.4. Keyword Snippet (LS-DYNA) Constraint

You can also insert additional LS-DYNA keywords by using the **Keyword Snippet (LS-DYNA)** constraint. Create a **Keyword Snippet (LS-DYNA)** object by inserting a Commands object from the context menu or Enviroment tab on the ribbon.

Note:

When inserted under Connections, the **Keyword Snippet (LS-DYNA)** object allows you to use contact types not supported by the LS-DYNA system. When inserted under the Environment object, it allows you to use any LS-DYNA keywords.

3.4.5. Bolt Pretension

This boundary condition applies a pretension load to a beam connection, typically to model a bolt under pretension.

Analysis Types

Bolt Pretension is specific to LS-DYNA and is not compatible with the Bolt Pretension feature of the Mechanical application.

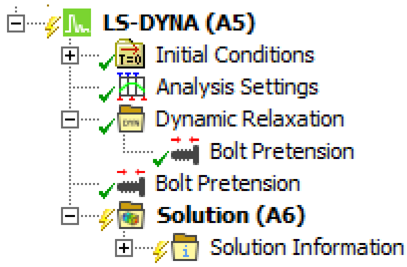
The Bolt Pretension can be either used during dynamic relaxation or during the explicit phase of the calculation.

A Bolt Pretension can be applied to either a Beam Connection or a Solid Body.

Boundary Condition Application

To apply a Bolt Pretension to a Beam Connection:

1. Right-click the **Environment** tree object or an active **Dynamic Relaxation Object** and select **Insert** → **Bolt Pretension**.



2. Set the **Scoping Method** to Beam Connection and then select the **Beam Connection**.

3. Specify the **Magnitude** of the loading.

Scope	
Scoping Method	Beam Connection
Beam Connection	Beam
Definition	
Magnitude	2 N

4. If the bolt pretension is used during the explicit phase, you need additionally an **Initialization End Time** to specify the termination of the loading.

Scope	
Scoping Method	Beam Connection
Beam Connection	Beam
Definition	
Magnitude	2 N
Initialization End Time	1E-05 s

To apply a Bolt Pretension to a Solid Body:

1. Right-click the **Environment** tree object or an active **Dynamic Relaxation Object** and select **Insert** → **Bolt Pretension**.

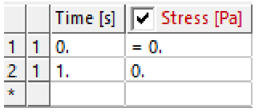
2. Set the **Scoping Method** to Geometry Selection or Named Selection and then select the Solid Body

Details of "Bolt Pretension"

Scope	
Scoping Method	Geometry Selection
Geometry Selection	1 Body
Definition	
Coordinate System	Global Coordinate System
Preload Stress	Tabular Data
Shear Stress Flag	Shear and Bending

3. Specify a **Coordinate System** to define the cutting plane. The cutting plane is centered on the origin of the selected Coordinate System and aligned with the X-Y plane.

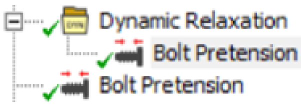
4. Define the pre-load stress as a function of time using the **Tabular Data** field, and define the type of Shear Stresses acting on the body with the **Shear Stress Flag**.



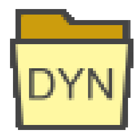
		Time [s]	Stress [Pa]
1	1	0.	= 0.
2	1	1.	0.
*			

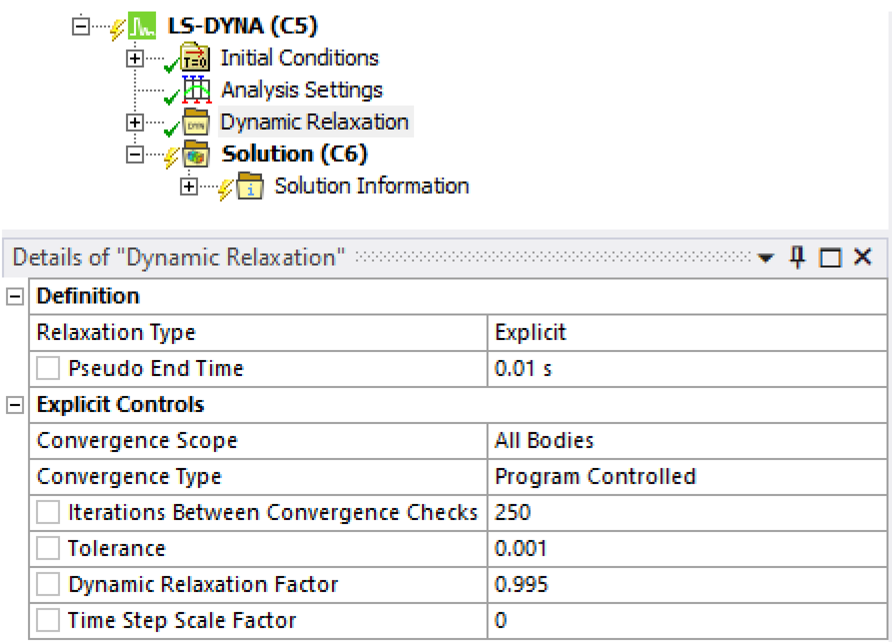
Note:

- The Bolt Pretension Load is not supported for a Full Restart.
- When a Bolt Pretension within the Dynamic Relaxation Folder and a Bolt Pretension under the LS-DYNA Transient Analysis are defined for the same Beam connection, only the last one defined is used in the analysis.



3.4.6. Dynamic Relaxation

The dynamic relaxation feature (available by clicking  on the LSDYNA Pre tab, or by right-clicking the LS-DYNA system and selecting Dynamic Relaxing from the Insert menu) provides preloading for explicit dynamics solutions in LS-DYNA. True dynamic relaxation (**Relaxation Type**: Explicit) allows an explicit solver to conduct a static analysis by increasing the damping until the kinetic energy drops to zero.

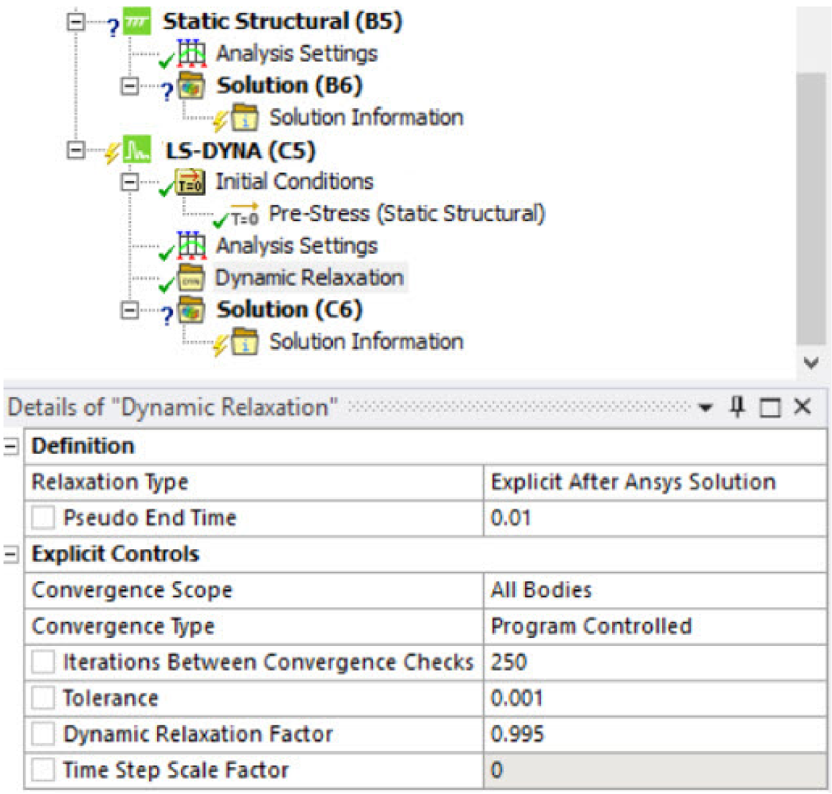


Definition	
Relaxation Type	Explicit
Pseudo End Time	0.01 s
Explicit Controls	
Convergence Scope	All Bodies
Convergence Type	Program Controlled
Iterations Between Convergence Checks	250
Tolerance	0.001
Dynamic Relaxation Factor	0.995
Time Step Scale Factor	0

The damping works by scaling nodal velocities by the **Dynamic Relaxation Factor** each time step until the ratio of current distortional kinetic energy to peak distortional kinetic energy (the convergence factor) falls below the convergence tolerance (**Tolerance**).

By default, the convergence is checked on the whole model. It can be restricted to a set of bodies by setting the **Convergence Scope** to Geometry Selection.

When the Ansys Implicit solver is used to provide the preload (**Relaxation Type**: Explicit After Ansys Solution), a slightly different approach is taken in that the stress initialization is based on a prescribed geometry (in other words, the nodal displacement results from the Implicit solution). In this case, the explicit solver only uses 101 time steps to apply the preload. In the former case, the solver will check the kinetic energy every 250 cycles (by default) until the kinetic energy from the applied preload is dissipated.



If the **Convergence Type** is set to Termination occurs at Pseudo End Time instead of Program Controlled, the termination of the dynamic relaxation occurs at the pseudo end time.

The **Time Step Scale Factor** enables you to scale the computed time step during dynamic relaxation.

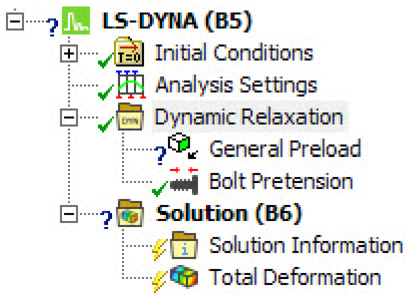
Alternatively, the LS-DYNA Implicit solver can be used to conduct a dynamic analysis to calculate the preloading.

An initial time step must be provided to start the nonlinear implicit transient analysis. The convergence of this Newton-Raphson analysis is controlled by the **Line Search** convergence tolerance and a **Displacement Convergence** tolerance.

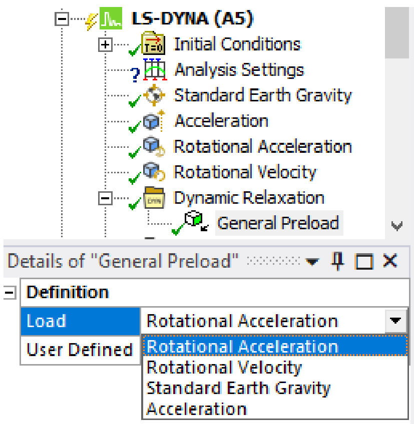
LS-DYNA supports all these methods, which occur in pseudo time before the transient portion of the analysis begins at time zero.

Preloading

Preloading works by specifying that a given load will be active during the dynamic relaxation, when the relaxation type is set to Explicit or Implicit.



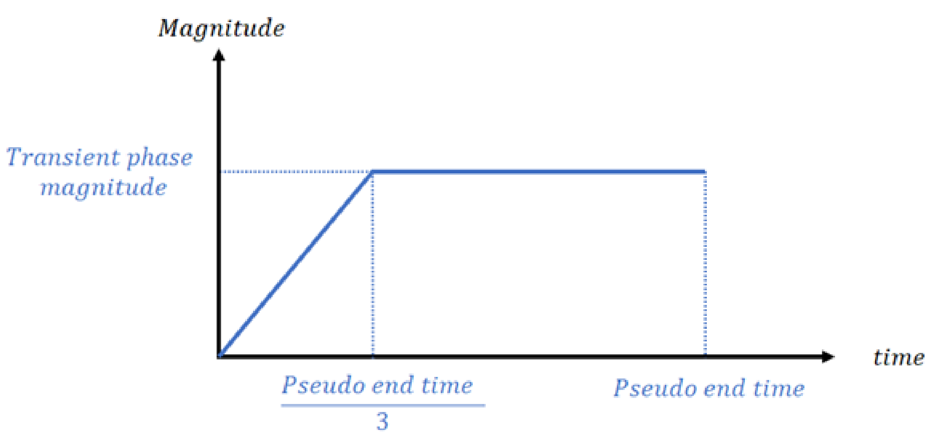
Currently, Acceleration, Standard Earth Gravity, Rotational Acceleration and Rotational Velocity can be specified as a **Load** in the **General Preload** object for dynamic relaxation.



Other Loads and Supports can be applied during dynamic relaxation through the use of an option on the Load/Support.

Preloading using General Preload

When the **User Defined** field is set to No, the load used during the dynamic relaxation phase is represented by the following curve:



For the boundary condition undergoing dynamic relaxation, an additional curve (*DEFINE_CURVE) is written defining the load magnitude during the dynamic relaxation phase. The standard curve shown above is applied only during the dynamic relaxation phase (preloading) as SIDR=1 in the *DEFINE_CURVE.

When the **User Defined** field is set to Yes, you can enter a customized curve for the dynamic relaxation phase by filling a data table.

When the load is user defined, the card *DEFINE_CURVE with the SIDR parameter set to 1 will be written to the input file. This curve is then used by the card representing the boundary condition to which the dynamic relaxation is applied.

Note:

Unlike other Dynamic Relaxation loads compatible with the General Preload object, both Rotational Velocity and Rotational Acceleration do not use any directional General Preload Scale Factor. This is related to limitations from the LS-DYNA card *LOAD_BODY_GENERALIZED that is used. It is still possible to apply these loads in specific directions, through the definition of the Coordinate System property in the Rotational Velocity or Rotational Acceleration objects themselves. Be mindful that the same preload curve will be applied to any loaded direction in these objects.

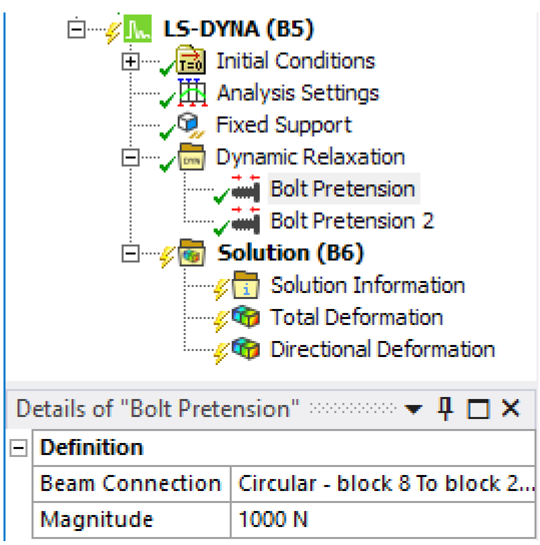
Preloading using a Load/Support Option

Dynamic relaxation can be defined in a simpler way for several loads (including Pressure, Force, Nodal Force, and Rigid Body Angular Velocity), and Supports that are defined so that they are not fixed. For these boundary conditions, the **Dynamic Relaxation Behavior** field is added to their Details panel. This field has three options:

- Normal Phase Only - The boundary condition is applied during the explicit solution only
- Dynamic Relaxation Only - The boundary condition is applied during preloading only
- Both - The boundary condition is applied in both phases

Preloading using Bolt Pretension

LS-DYNA also enables preloading of Beam Connections through the Bolt Pretension.

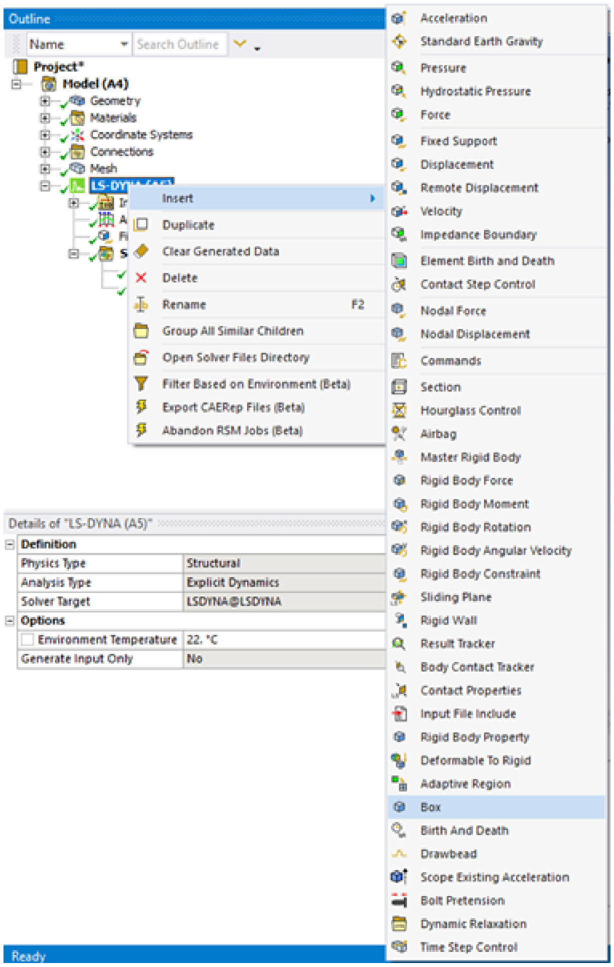


3.4.7. Bounding Box

A *Box* is an object that can be used to define a behavior for a set of nodes. For instance, it can be used to specify that SPH computations occur for particles inside the box.

3.4.7.1. Defining a Box

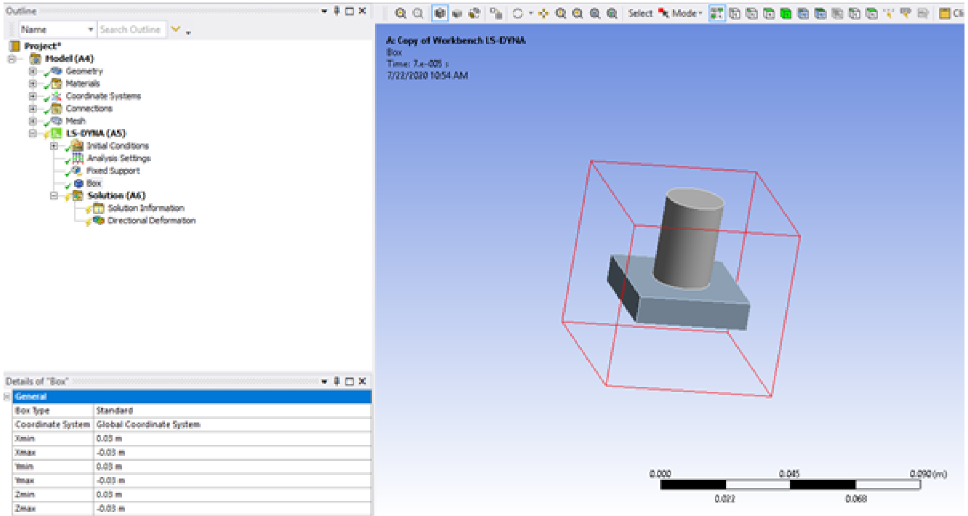
To insert a box in an LS-DYNA analysis, right-click the system and select **Insert** → **Box**.



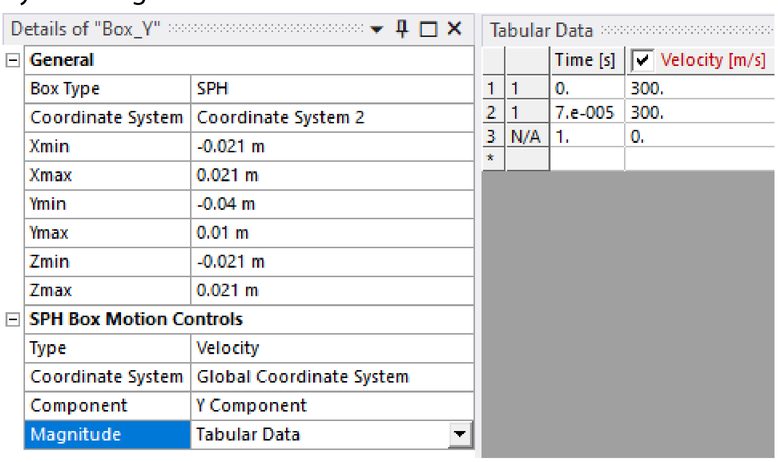
Two types of boxes exist: Standard and SPH. The Standard box is a stationary box. At this time, its function is to limit the SPH or SPG computation to particles that are within the box at the beginning of the simulation.

For the Standard box, the input parameters are:

- The boundary of the box defined by the coordinates of two diagonally opposite corner points: **Xmin**, **Xmax**, **Ymin**, **Ymax**, **Zmin**, **Zmax**.
- The **Coordinate System** that defines the orientation of the box. You can define a customized coordinate system (new origin and new orientation) then select that coordinate system to assign it to the box.



The SPH box is a standard box that can move over time and is used only for SPH calculations. The **SPH Box Motion Controls** section in the Details panel allows you to specify the motion **Type** as a Displacement or Velocity. The direction of the box motion is defined by the **Coordinate System** and the **Component**. You can define time-dependent velocity or displacement for the box motion by entering tabular data.



Note:

For SPH analyses, the box must include the contact surface region between the particles and the Lagrangian region in order to properly determine the mechanical behavior.

3.4.7.2. Using a Box in the Analysis

You can define any number of Standard and SPH boxes in an analysis. In conjunction with the **Use Box** analysis setting, the following rules govern the solver's use of the boxes defined in the analysis.

If Use Box is set to No

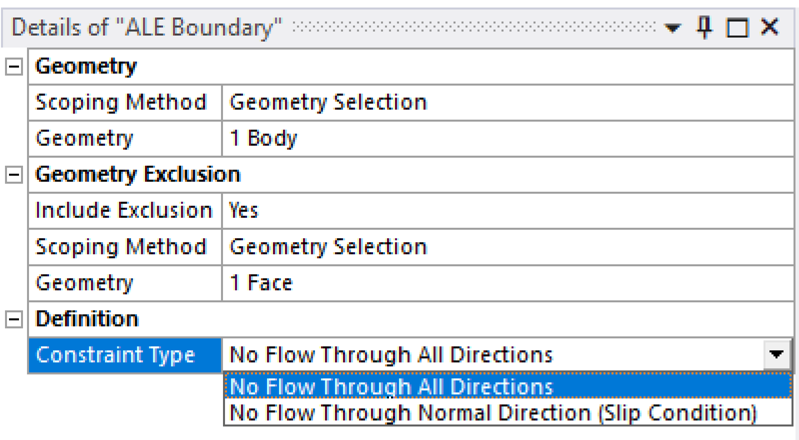
- Any Standard boxes defined in the analysis will not be used. However, if you have an unsuppressed SPH box, then the calculation will occur with respect to the SPH box. If you have multiple unsuppressed SPH boxes, then the solver will consider the first defined SPH box.

If Use Box is set to Yes

- If you only have Standard boxes defined, you can choose a Standard box from the **Box Name** drop down.
- If you only have SPH boxes defined, you don't need to set the Box Name because the first unsupressed defined SPH box will be used. An information message will be output indicating the Box that is used.
- If you have both Standard and SPH boxes defined, if you choose a Standard box from the **Box Name** drop-down, the solver will ignore that box and will use the first unsuppressed SPH box that is defined. When an SPH box is used, an information message will pop up to indicate the Box used by the solver.

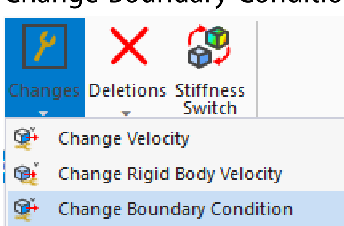
3.4.8. ALE Boundary

The **ALE Boundary** object can be applied to bodies with **Reference Frame** set to S-ALE Domain, or contained in a section definition with an ALE element formulation. When an ALE Boundary object is included in the analysis, an *ALE_ESSENTIAL_BOUNDARY card is written. Two constraint types are available, which will be applied to the external faces of the main geometry selection except for any faces defined in the exclusion scoping.



3.4.9. Change Boundary Condition

The **Change Boundary Condition** load can be inserted in an LS-DYNA Small Restart analysis and allows you to change the definition of an existing *DEFINE_CURVE (p. 204) keyword in the *input.k* file. The Change Boundary Condition load can be found on the LSDYNA Small Restart tab of the ribbon.



The *DEFINE_CURVE keyword is written to the input file when tabular data is provided for a load. The Change Boundary Condition object is needed because in an LS-DYNA analysis, the end time of the tabular data defined in the analysis cannot exceed the end time of the analysis itself. For example, a Displacement Component is defined by a Tabular Data in an initial LS-DYNA analysis and you want to continue with a Small Restart. You cannot change the values of the Displacement Component for the Restart analysis (the object is unavailable in the Small Restart). The Change Boundary Condition object gives you this ability.

The load will redefine a *DEFINE_CURVE that already exists in the initial (or one of the previous) analyses. The tabular data you enter must correspond to the size of the table in the existing *DEFINE_CURVE. The *DEFINE_CURVE to be modified might have been generated by the Tabular Data defining one of the degrees of freedom of a boundary condition (currently, only **Displacement** and **Remote Displacement** are supported).

When you insert a **Change Boundary Condition** object, the **Location Method** will be set to Boundary Condition.

You will see the following fields in the Details panel. Once you choose a Boundary Condition from the drop-down list, fields will appear corresponding to the components using tabular data in that boundary condition.

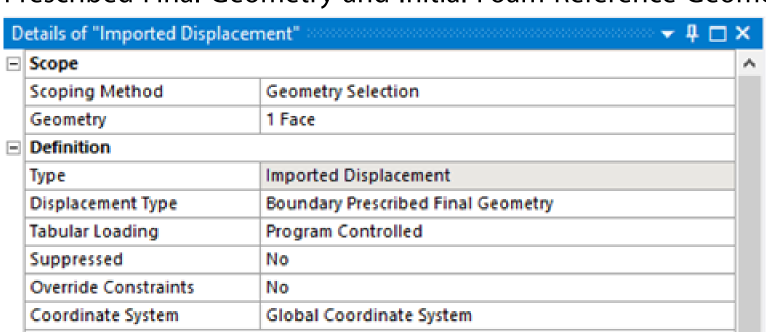
Field	Input Values	Description
Location Method	Boundary Condition	Specifies that the tabular data to be redefined is in a boundary condition
Boundary Condition	Drop-down with available boundary conditions	Specifies which boundary condition needs to have tabular data redefined
Component	X/Y/Z Component	Specifies which degree of freedom will have the tabular data in its *DEFINE_CURVE redefined
Curve	Tabular Data	Allows redefinition of the data in the *DEFINE_CURVE associated with the component

3.4.10. Importing External Loads

The External Data system allows you to import external loads into an LS-DYNA analysis. Currently, the loads supported for LS-DYNA analyses are Imported Pressure and **Imported Displacement**.

3.4.10.1. Imported Displacement

When an **Imported Displacement** is added to an LS-DYNA analysis, the keyword written to the input file is determined by the **Displacement Type** field. The two keyword options are Boundary Prescribed Final Geometry and Initial Foam Reference Geometry.



When selecting the displacement for each component in the Data View worksheet, the Free option available for other types of analyses is not shown as it is not applicable when using these LS-DYNA keywords.

Imported Displacement

	X Component (m)	Y Component (m)	Z Component (m)	Analysis Time (s)	Scale
1	File1:Displacement1	Fixed	Fixed	0.005	0.5
*					

When the **Displacement Type** is set to Initial Foam Reference Geometry the **Tabular Loading** field is hidden. The rate at which the displacement is applied is determined by the solver in this case.

Imported Displacement

	X Component (m)	Y Component (m)	Z Component (m)	Scale
1	File1:Displacement1	Fixed	Fixed	0.5
*				

The **Displacement Type** Initial Foam Reference Geometry is only applicable to a subset of materials. Refer to the description of *Initial_Foam_Reference_Geometry in LS-DYNA Keyword User's Manual Volume I for a list of the supported material types.

Note:

When the Imported Displacement is scoped to a body with a supported material, the reference geometry flag on the material card is automatically set. If the material is defined using a command snippet, the reference geometry flag is not automatically set and the value given in the snippet is written directly to the input file.

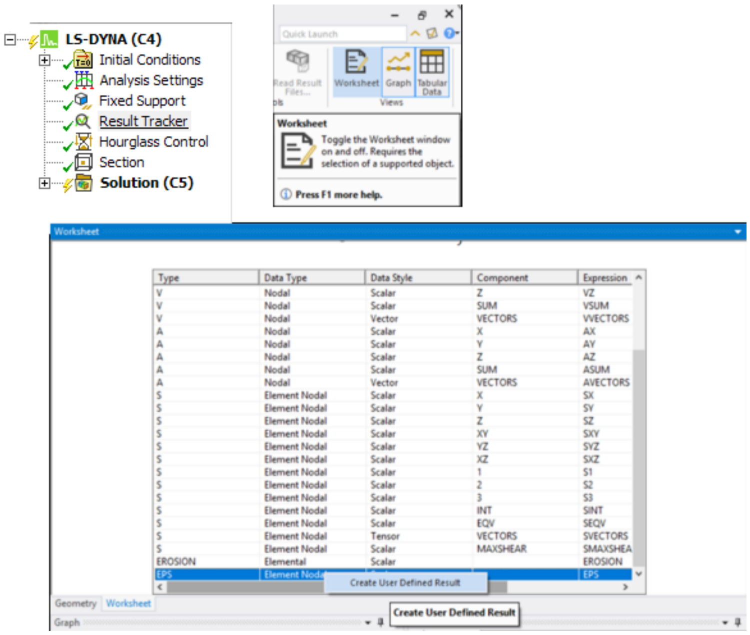
3.5. Accessing Results

You can find information about Results Processing in Review Results.

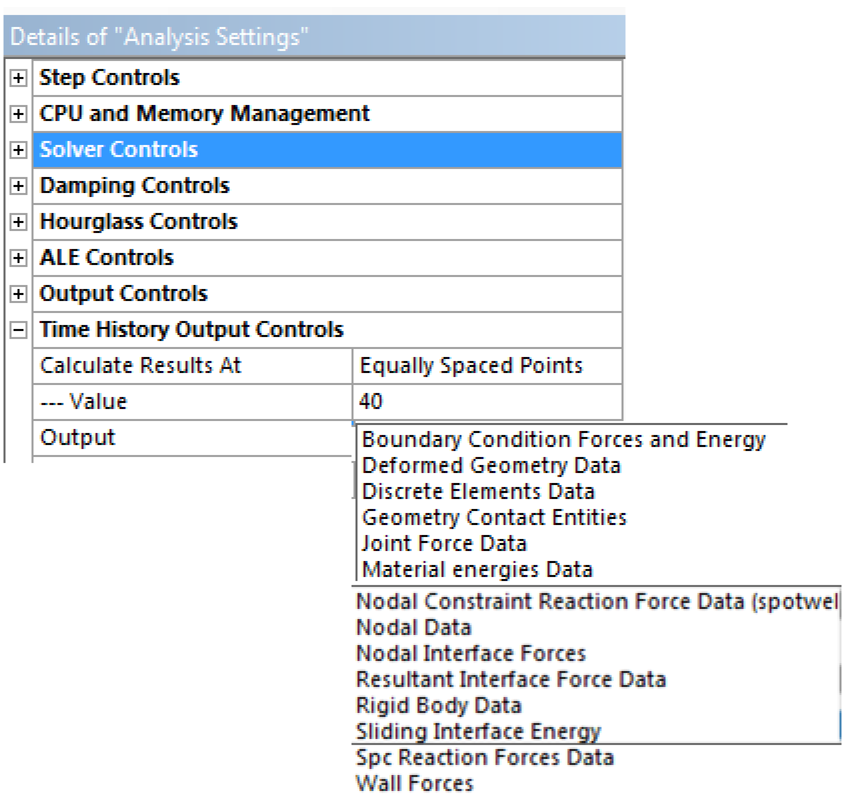
To collect Nodal Data, insert one or more Result Trackers under Solution prior to running the simulation. Result trackers must be scoped to a node.

Stress and Plastic Strain are written by default (can be suppressed) but Strain is not written by default and you must select it if desired.

LS-DYNA keeps track of Total Strain and Plastic Strain. Elastic Strain will always show as 0. To plot elastic plastic strain or elastic strain, click **Solution**, then click Worksheet from the Home tab and select Available Solution Quantities within the Worksheet. Right-click the expression and select Create User Defined Result.



Time History Outputs (ASCII files) can be generated. All the files shown below can be created using the LS-DYNA system. Note however that some data in the ASCII files currently cannot be viewed.



To view Time History Outputs select the ones desired from the ASCII drop-down menu.

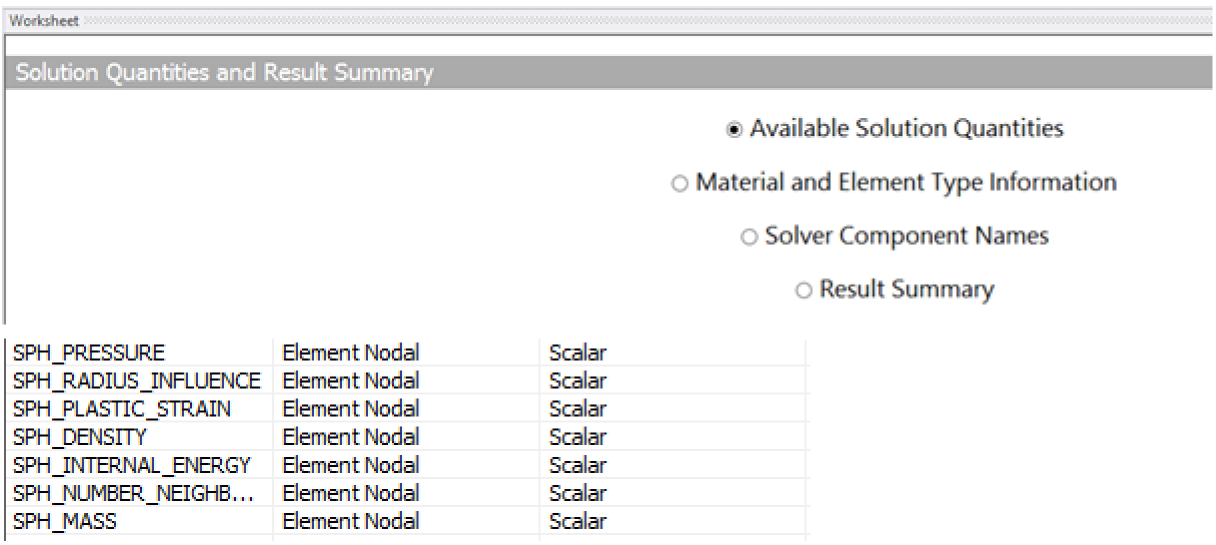
Using LS-DYNA terminology, the following results can be viewed within LS-DYNA:

- GLSTAT (Global Data : Kinetic Energy, Hourglass, ...).

- BNDOUT (Boundary Conditions Data).
- RCFORC (Contact Forces Data).
- SPCFORC (Reaction Forces on Boundary Conditions) using *BOUNDARY_SPC (Fixed Support) to view Reaction Force.
- MATSUM (Body Data).
- NODOUT (Nodal Data) Trackers must be defined during pre-processing for the nodal data to be available during Results processing.

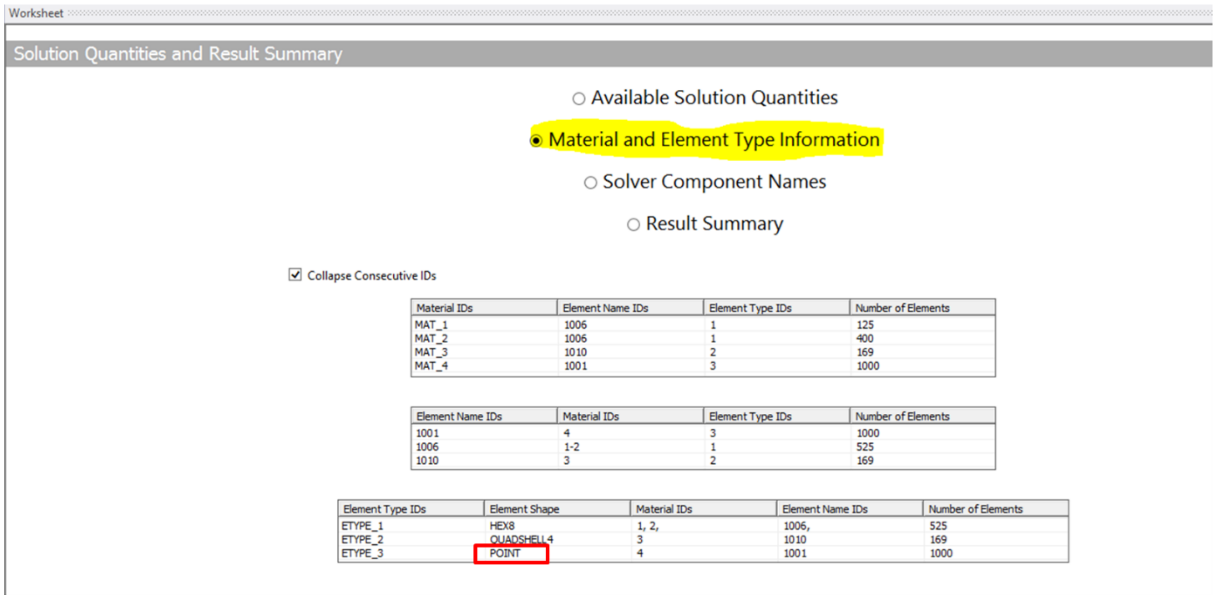
3.5.1. Post-processing for Solver-Created SPH Elements

When an SPH body or an Adaptive Solid to SPH object is included in the model, several SPH-specific results are available as user defined variables. These results can be found in the worksheet as shown in the following images.



When using an Adaptive Solid to SPH object in the model, SPH particles are constructed inside the solver during the solution. To obtain results for these created SPH particles, do the following:

1. Access the worksheet, then select Material and Element Type Information as shown in the following figure:



2. Locate the POINT element shape as depicted in the previous figure. By right-clicking the corresponding line, you can insert a total deformation result or any user-defined result. Remember that the available user-defined results can be viewed by selecting Available Solution Quantities on the worksheet.

3.5.2. Post-processing for Subset of Parts

LS-DYNA allows you to store the results in a different file, which is separate from the main result file (`d3plot`). This file contains similar information, but is specifically for selected bodies or parts of the model. Since this file only contains results for a few bodies, the output frequency can be increased and more time steps can be stored in the result file.

Activate the output of the `d3part` file in Mechanical by setting **Selective Output** in **Output Controls** to Equally Spaced Points, or Time. If set to Equally Spaced Points, the **Value** field should be set to the number of time steps. If set to Time, the **Value** field should define a time interval between each output.

You must specify a named selection containing the bodies for inclusion in the `d3part` file using the **Selective Output Scope** field.

To display a particular result from the set of bodies selected for the `d3part` output, set **Result File** to `d3part`.

Details of "Equivalent Stress"

Scope	
Scoping Method	Geometry Selection
Geometry	All Bodies
Definition	
Type	Equivalent (von-Mises) Stress
By	Time
Display Time	Last
Separate Data by Entity	No
Result File	d3part
Calculate Time History	Yes
Identifier	
Suppressed	No
Integration Point Results	
Display Option	Averaged
Average Across Bodies	No
Results	
Minimum	
Maximum	
Average	
Minimum Occurs On	
Maximum Occurs On	
Information	

3.5.3. Capture Maximum Stress during Calculation

Explicit calculations often require thousands of calculation cycles. The result files typically contain a few dozens of these cycles.

Note:

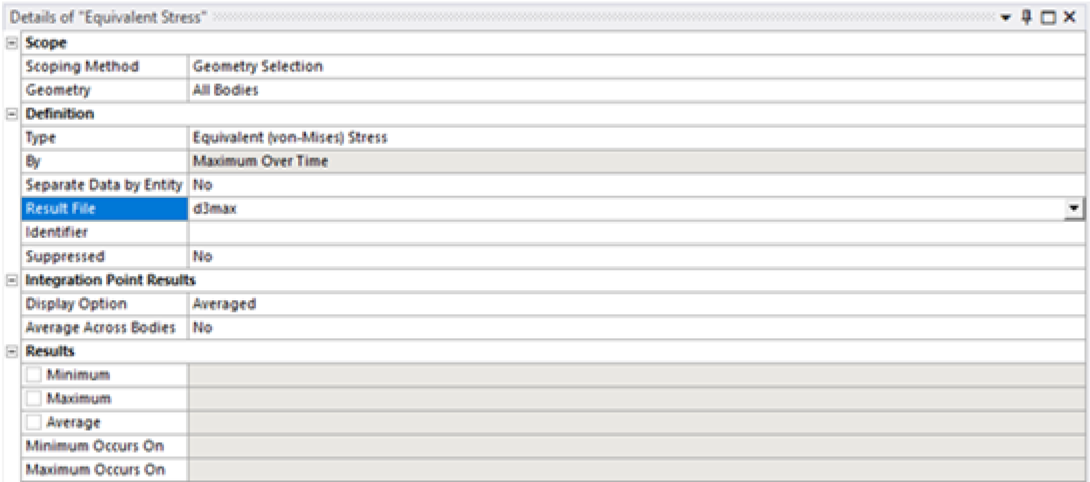
It is not possible to determine the maximum stress during the calculation. This could occur during unsaved cycles.

When you set **Capture Maximum Stress Over Time** in the Analysis Settings Output Controls category to Yes, LS-DYNA saves the maximum stress that occurred during the calculation. At each **Capture Time**, LS-DYNA compares the maximum stress to the previously saved value at each element of the mesh, and updates it if it increases.

Capture Maximum Stress Over Time	Yes
Capture Time	

The results are saved in the `d3max` file.

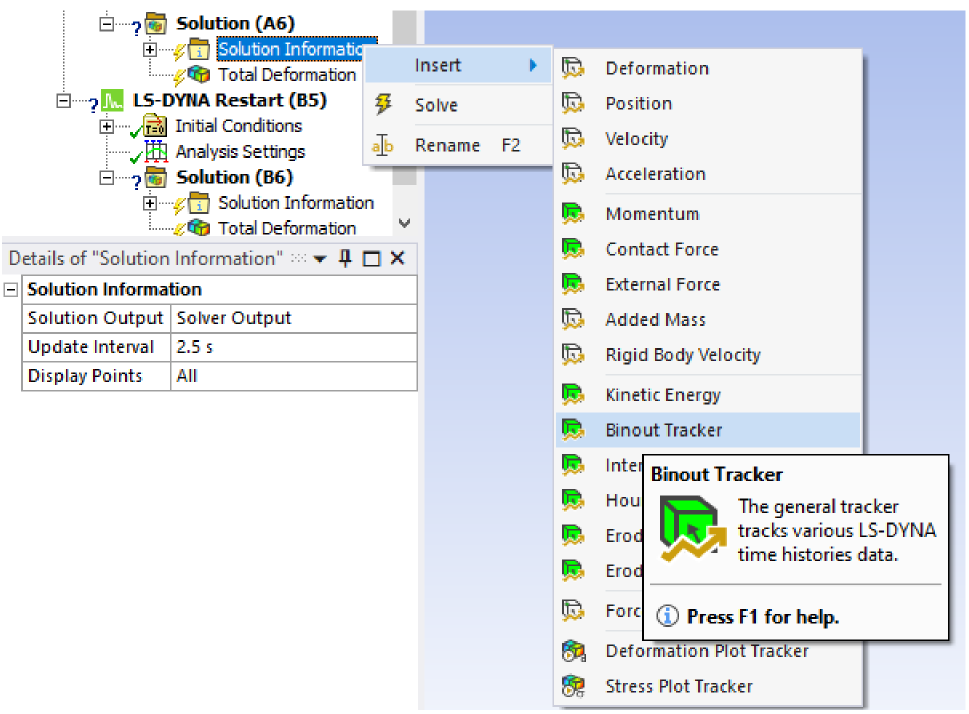
`D3max` results are only available for stress results. To display them, set **By** in the **Definition** category of the stress result to Maximum Over Time, and set **Result File** to `d3max`.



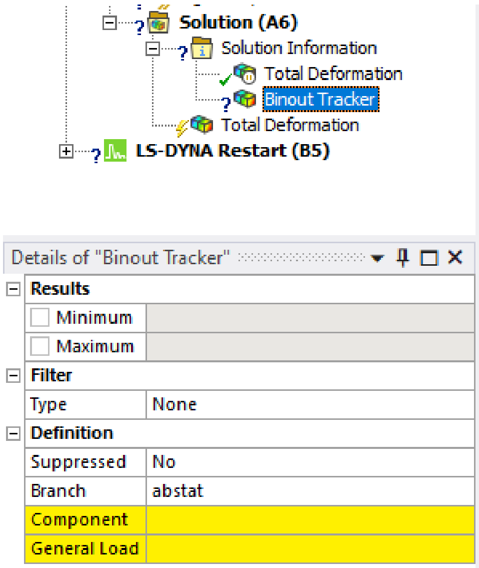
3.5.4. The Binout Tracker

The Binout Result Tracker provides extensive access to the various results stored in the LS-DYNA result file `binout`.

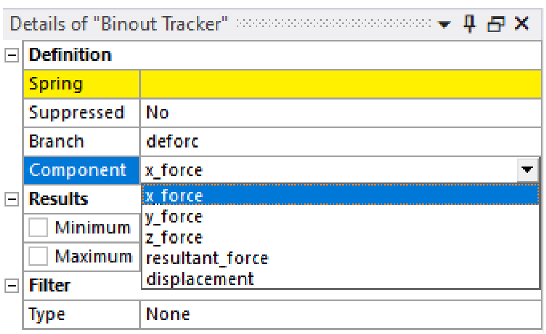
To add a Binout Tracker, right-click the Solution Information object in the tree and select **Insert** → **Binout Tracker**.



`Binout` is a binary file using the LSTC Data Archival Format or LSDA. This format mimics a Unix file system. Each result has the structure of a directory. It is called a **Branch** in the LSDA terminology. Each branch can have several subdirectories (sub-branches).



Each branch can have a number of components that allow you to access and plot the various results available for that branch.



The dbfsi data is available when **ALE FSI Tracker Data** is defined under Output Controls. FSI data is created for each Coupling object with the specified output frequency.

ALE FSI Tracker Data	Time
ALE FSI Tracker Data (DT)	3.45E-08 s

Mechanical supports the following Binout branches (Mechanical does not currently support all available components for each branch):

abstat : Airbags results.

bndout: Boundary condition forces and energy results.

dbfsi: Fluid structure interaction data.

defgeo: Deformed geometry file results.

deforc: Discrete spring and discrete damper results.

elout: Elemental results. Mechanical defaults to and only supports **elout** beam or shell IP1 (Integration Point 1) results.

glstat: Global results.

jntforc: Joint results.

matsum: Part energies results.

nodout: Nodal translational motion results.

rbdout: Motion of rigid bodies in global and local coordinate systems.

rcforc: Resultant contact interface forces.

rwforc: Rigidwall forces.

secforc: Cross section forces.

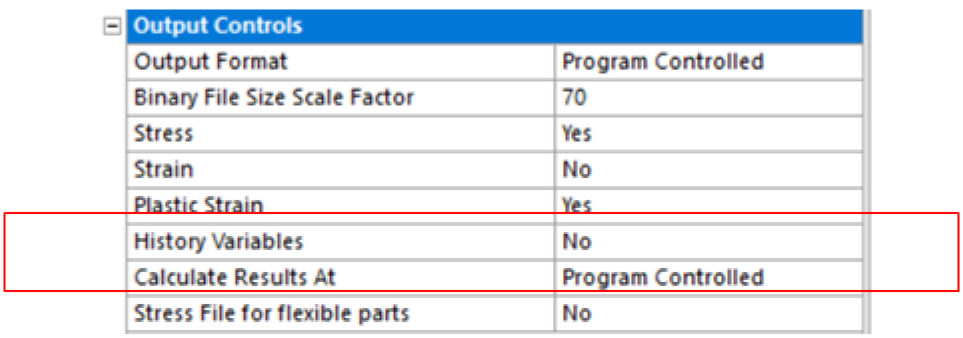
sleout: Contact interface energies.

spcforc: Single point constraint reaction forces.

3.6. History Variable Output

Material models in LS-DYNA use history variables that are specific to the material model. For most materials, six variables are reserved for the Cauchy stress components and one for the effective plastic strain, but some models have many more. The extra history variables may include properties like strain energy density and strain rate, and may also include information like material direction cosines and scale factors.

The output of these extra history variables is defined in the **Output Controls** section of the Analysis Settings Details.



Output Controls	
Output Format	Program Controlled
Binary File Size Scale Factor	70
Stress	Yes
Strain	No
Plastic Strain	Yes
History Variables	No
Calculate Results At	Program Controlled
Stress File for flexible parts	No

If you set **History Variables** to Yes (the default is No), LS-DYNA automatically sets the number of history variables for shell and solid elements in the model. It calculates the value depending on the materials and on the type of bodies present in the model.

Depending on the material assigned to your body, you will have history variables available in the Solution Worksheet and can evaluate them as a User Defined Result.

Type	Data Type	Data Style	Component	Expression	Output Unit
U	Nodal	Scalar	SUM	USUM	Displacement
U	Nodal	Vector	VECTORS	UVECTORS	Displacement
V	Nodal	Scalar	X	VX	Velocity
V	Nodal	Scalar	Y	VY	Velocity
V	Nodal	Scalar	Z	VZ	Velocity
V	Nodal	Scalar	SUM	VSUM	Velocity
V	Nodal	Vector	VECTORS	VVECTORS	Velocity
A	Nodal	Scalar	X	AX	Acceleration
A	Nodal	Scalar	Y	AY	Acceleration
A	Nodal	Scalar	Z	AZ	Acceleration
A	Nodal	Scalar	SUM	ASUM	Acceleration
A	Nodal	Vector	VECTORS	AVECTORS	Acceleration
S	Element Nodal	Scalar	X	SX	Stress
S	Element Nodal	Scalar	Y	SY	Stress
S	Element Nodal	Scalar	Z	SZ	Stress
S	Element Nodal	Scalar	XY	SXY	Stress
S	Element Nodal	Scalar	YZ	SYZ	Stress
S	Element Nodal	Scalar	XZ	SXZ	Stress
S	Element Nodal	Scalar	1	S1	Stress
S	Element Nodal	Scalar	2	S2	Stress
S	Element Nodal	Scalar	3	S3	Stress
S	Element Nodal	Scalar	INT	SINT	Stress
S	Element Nodal	Scalar	EQV	SEQV	Stress
S	Element Nodal	Tensor	VECTORS	SVECTORS	Stress
S	Element Nodal	Scalar	MAXSHEAR	SMAXSHEAR	Stress
EROSION	Elemental	Scalar		EROSION	No Units
EPS	Element Nodal	Scalar		EPS	Strain
THICKNESS	Element Nodal	Scalar		THICKNESS	Displacement
BACKSTRESS	Element Nodal	Scalar	X	BACKSTRESSX	Stress
BACKSTRESS	Element Nodal	Scalar	Y	BACKSTRESSY	Stress
BACKSTRESS	Element Nodal	Scalar	Z	BACKSTRESSZ	Stress
BACKSTRESS	Element Nodal	Scalar	XY	BACKSTRESSXY	Stress
BACKSTRESS	Element Nodal	Scalar	YZ	BACKSTRESSYZ	Stress
BACKSTRESS	Element Nodal	Scalar	XZ	BACKSTRESSXZ	Stress
BACKSTRESS	Element Nodal	Scalar	1	BACKSTRESS1	Stress
BACKSTRESS	Element Nodal	Scalar	2	BACKSTRESS2	Stress
BACKSTRESS	Element Nodal	Scalar	3	BACKSTRESS3	Stress
BACKSTRESS	Element Nodal	Scalar	INT	BACKSTRESSINT	Stress
BACKSTRESS	Element Nodal	Scalar	EQV	BACKSTRESSEQV	Stress
BACKSTRESS	Element Nodal	Tensor	VECTORS	BACKSTRESSVECTORS	Stress
BACKSTRESS	Element Nodal	Scalar	MAXSHEAR	BACKSTRESSMAXSHEAR	Stress
SOLIDHIST_1	Element Nodal	Scalar		SOLIDHIST_1	No Units
SOLIDHIST_2	Element Nodal	Scalar		SOLIDHIST_2	No Units
SOLIDHIST_3	Element Nodal	Scalar		SOLIDHIST_3	No Units
SOLIDHIST_4	Element Nodal	Scalar		SOLIDHIST_4	No Units
SOLIDHIST_5	Element Nodal	Scalar		SOLIDHIST_5	No Units

The number of extra history variables is dependent on the material model used. The following table shows the history variables for the equation of state/material combinations supported by LS-DYNA.

Materials Supported in LSDYNA	**Corresponding Mat Number**	**Var No.**	**Shells (and Thick Shells Types 1,2,6)**	**Var No.**	**Solids (and Thick Shell Types 3,5,6**
*MAT_3-PARAMETER_BARLAT	*MAT_036				
		1	reverse of old yield stress		
		3	cosine(alpha)		
		4	- sine(alpha		

Materials Supported in LSDYNA	Corresponding Mat Number	Var No.	Shells (and Thick Shells Types 1,2,6)	Var No.	Solids (and Thick Shell Types 3,5,6
		6	effective strain rate		
		7	current yield stress		
		8	current hardening slope		
		9	back stress component alpha_11		
		10	back stress component alpha_22		
		11	back stress component alpha_12		
*MAT_BARLAT_ANISOTROPIC_PLASTICITY	*MAT_033	NA			
*MAT_ADD_EROSION	NA	NA			
*MAT_ARRUDA_BOYCE_RUBBER	*MAT_127	NA			
*MAT_BLATZ-KO_RUBBER	*MAT_007	NA		1..9	deformation gradient
*MAT_ELASTIC	*MAT_001	eqp	volumetric strain	eqp	volumetric strain
*MAT_ENHANCED_COMPOSITE_DAMAGE	*MAT_054-055				
	(MAT_054)	1	flag for longitudinal tensile failure mode		
	(MAT_054)	2	flag for longitudinal compressive failure mode		
	(MAT_054)	3	flag for transverse tensile failure mode		
	(MAT_054)	4	flag for transverse compressive failure		
	(MAT_054)	5	mode		
	(MAT_054)	6	total failure		
	(MAT_054)	8	damage parameter (SOFT)		

Materials Supported in LSDYNA	Corresponding Mat Number	Var No.	Shells (and Thick Shells Types 1,2,6)	Var No.	Solids (and Thick Shell Types 3,5,6
	(MAT_054)	9	cosine(alpha)		
	(MAT_054)	10	- sine(alpha)		
	(MAT_054)	11	local strain a-direction		
	(MAT_054)	12	local strain b-direction		
	(MAT_054)	16	local shear strain (ab-plane) transverse shear damage		
*MAT_LAMINATED_COMPOSITE_FABRIC	*MAT_058				
		1	damage in longitudinal direction		
		2	damage in transverse direction		
		3	damage in shear		
		6	cosine(alpha		
		7	-sine(alpha		
		8	total failure flag (1=not failed, 0=failed)		
		10	local strain a-direction		
		11	local strain b-direction		
		12	local shear strain (ab-plane)		
		15	effective strain		
		16	local strain c-direction		
		17	local transverse shear strain (bc-plane)		
		18	local transverse shear strain (ca-plane)		

Materials Supported in LSDYNA	Corresponding Mat Number	Var No.	Shells (and Thick Shells Types 1,2,6)	Var No.	Solids (and Thick Shell Types 3,5,6
		19	damage variable for transverse shear behavior		
		20	strain-rate in a-direction		
		21	strain-rate in b-direction		
		22	strain-rate in ab-plane		
*MAT_FLD_TRANSVERSELY_ANISOTROPIC	*MAT_039	NA			
*MAT_HYPERELASTIC_RUBBER	*MAT_077_H	NA			
*MAT_JOHNSON_COOK	*MAT_015				
		1	failure value	5	temperature change
		3	current pressure cutoff	6	JC damage parameter
		4	JC damage parameter		
		5	temperature change		
		6	JC failure strain		
*MAT_MODIFIED_PIECEWISE_LINEAR_PLASTICITY	*MAT_123				
		1	effective strain rate	1	effective strain rate
		6	thinning strain	7	major principal strain
		7	major in plane strain		
*MAT_OGDEN_RUBBER	*MAT_077_O	NA			
*MAT_ORTHOTROPIC_ELASTIC, *MAT_ANISOTROPIC_ELASTIC	*MAT_002				
		eqp	stiffness component C11	1..9	deformation gradient
		1	stiffness component C12		
		2	stiffness component C13		
		3	stiffness component C14		

Materials Supported in LSDYNA	Corresponding Mat Number	Var No.	Shells (and Thick Shells Types 1,2,6)	Var No.	Solids (and Thick Shell Types 3,5,6
		4	stiffness component C22		
		5	stiffness component C23		
		6	stiffness component C24		
		7	stiffness component C33		
		8	stiffness component C34		
		9	stiffness component C44		
		10	stiffness component C55		
		11	stiffness component C56		
		12	stiffness component C66		
*MAT_PIECEWISE_LINEAR_PLASTICITY	*MAT_024				
		1	effective strain rate (VP=0); effective plastic strain rate (VP=1)	1	effective strain rate (VP=0); effective plastic strain rate (VP=1)
		4	current hardening slope	4	current hardening slope
		5	current yield stress Remark: 4 and 5 apply only to non-strain-rate option.	5	current yield stress Remark: 4 and 5 apply only to non-strain-r option.
*MAT_PLASTIC_KINEMATIC	*MAT_003				

Materials Supported in LSDYNA	Corresponding Mat Number	Var No.	Shells (and Thick Shells Types 1,2,6)	Var No.	Solids (and Thick Shell Types 3,5,6
		1	back stress component xx	1	back stress component xx
		2	back stress component yy	2	back stress component yy
		3	back stress component xy	3	back stress component xy
		4	back stress component yz	4	back stress component yz
		5	back stress component zx	5	back stress component zx
*MAT_POWER_LAW_PLASTICITY	*MAT_018	NA			
*MAT_RATE_SENSITIVE_POWERLAW_PLASTICITY	*MAT_064	NA			
*MAT_RIGID	*MAT_020	NA			
*MAT_SIMPLIFIED_JOHNSON_COOK	*MAT_098				
		1	max(0,ln(normalized eff strain rate)) (VP=0) failure flag (VP=1)		
*MAT_SPRING_INELASTIC	*MAT_S08	NA			
*MAT_SPRING_NONLINEAR_ELASTIC	*MAT_S04	NA			
*MAT_TRANSVERSELY_ANISOTROPIC_ELASTIC_PLASTIC	*MAT_037				
		2	current hardening slope		
		3	current yield stress		
*MAT_LOW_DENSITY_FOAM	*MAT_057	NA			

The following table shows the history variables for equation of state/material combinations supported by LS-DYNA.

Equation of State	Material	Var No.	Var Description
*EOS_JWL (EOS_002)	*MAT_HIGH_EXPLOSIVE_BURN (*MAT_008)	1	Internal Energy
		2	Bulk Viscosity

Equation of State	Material	Var No.	Var Description
		3	Volume
		4	Burn Fraction
		5	Afterburn Energy

Note:

For User Defined Results with the **Output Unit** value of No Units, the unit system used is the one defined in the **Unit System** field in the Analysis Settings Solver Controls section. For User Defined Results with an **Output Unit** value other than No Units, the unit system used is the one defined in the Mechanical Home tab.

3.7. Special Analysis Topics

This section covers special workflows and solution methods information. Some of the topics here include:

- Descriptions of how the LS-DYNA system can interact with other Workbench systems.
- ALE Workflow (p. 79)
- SPH (p. 83) and Incompressible SPH (p. 84) Workflows
- SPG and ISPG Workflows (p. 88)
- Multiple Case Workflow (p. 101)

3.7.1. Importing the Results of a Thermal Analysis

A Steady-State or Transient Thermal analysis system can be directly linked to an LS-DYNA system to import the calculated temperatures and provide thermal stress. For information on setting up this transfer, follow the description of how it is done for a structural system in Thermal-Stress Analysis in the *Mechanical User's Guide*

Note:

- Imported temperatures cannot be scoped to beams.
- LS-DYNA requires the definition of the Instantaneous Thermal Expansion Coefficient on materials to calculate thermal deformations.

3.7.2. Importing the Results of an FSI Thermal Analysis

A CFD analysis system can be directly linked to an LS-DYNA system to import the Temperature results from a heat transfer CFD analysis as body temperature loads. For information on setting up this transfer, follow the description of how it is done for a structural system in Using Imported Loads for One-Way FSI in the *Mechanical User's Guide*.

3.7.3. Importing the Pressure Results of an FSI Analysis

A CFD analysis system can be directly linked to an LS-DYNA system to import the Pressure results from a CFD analysis as imported pressures loads. For information on setting up this transfer, follow the description of how it is done for a structural system in Using Imported Loads for One-Way FSI in the *Mechanical User's Guide*.

3.7.4. Thermal Workflow

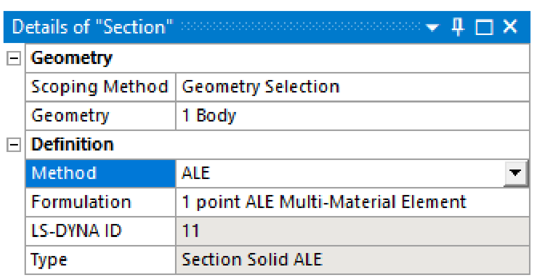
When the Solver Type analysis setting is set to Coupled Structural Thermal Analysis, the LS-DYNA system in Mechanical supports thermal boundary conditions and results. The addition of these boundary conditions enables strongly coupled structural-thermal calculations, in support of simulations like frictional heating, battery abuse, or other analyses where mechanical and thermal solutions affect each other strongly.

The LS-DYNA system provides support for the following thermal loads in Mechanical: Temperature, Convection, and Heat Flux. These loads are implemented, respectively, with the keywords *BOUNDARY_TEMPERATURE (p. 207), *BOUNDARY_CONVECTION (p. 207), and *BOUNDARY_FLUX (p. 208). The system also supports Temperature and Heat Flux results, and the Temperature Tracker. Refer to the object descriptions for usage notes for LS-DYNA systems.

3.7.5. ALE Workflow

LS-DYNA provides two types of ALE solvers:

- The *unstructured ALE solver*, which can be defined within Mechanical by assigning an ALE Method type. The unstructured mesh is created within Mechanical and is passed directly to the solver.



Details of "Section"

Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Method	ALE
Formulation	1 point ALE Multi-Material Element
LS-DYNA ID	11
Type	Section Solid ALE

- The *structured ALE (S-ALE) solver* can be invoked by creating a body with rectilinear box geometry and a **Reference Frame** set to S-ALE Domain. The default material for the domain is associated directly with the body. Additional bodies with **Reference Frame** set to S-ALE Fill, are filled into the S-ALE Domain. This is achieved by first meshing the S-ALE Fill bodies using the standard Mechanical meshing process. The surface meshes of these bodies are passed to the solver as dummy rigid shell bodies and are used to define the material with the S-ALE Domain. Once the filling is complete, the dummy rigid shell bodies have no further effect on the simulation. For more information, see the Initial Volume Fraction Geometry (p. 150) keyword.

  The S-ALE Mesh Object (p. 32) is used to define mesh parameters and view the surface mesh for the S-ALE mesh that will be created within the solver.

Several objects are available for use in the ALE workflow:

- The Coupling (p. 28) object which is added under connections, enables the setup of ALE-Lagrange interactions.
- The ALE Boundary Object (p. 62) allows you to define stick and slip boundary conditions on the external faces of the ALE domain.

Note:

- The mesh created within Mechanical is not directly used for the S-ALE domain. By default, the bounding box of the geometry and the number of elements generated by the Mechanical meshing process are used as parameters to define the size and density of a uniform rectilinear mesh that will be created by the solver. Further details are given in S-ALE Mesh (p. 32).
- Results for S-ALE bodies are not displayed on the original mesh. Instead, a mesh is re-constructed for each material associated with the original body to which the result object is scoped.
- The reconstruction of the mesh is approximate and includes:
  - Finding the exterior surface of each material in its current location in the S-ALE domain. This is achieved by forming an isosurface on the volume fraction of each material in a cell (at 50%).
  - Filling the interior of the material with cells from the S-ALE domain that are completely inside the material.
  - Reconstructing an unstructured mesh for any gaps between the exterior surface and interior cells.
- When ALE bodies are included in the model, several ALE-specific result variables are available as user defined variables: Density, Volume Fraction, Dominant Material, and Species Mass.

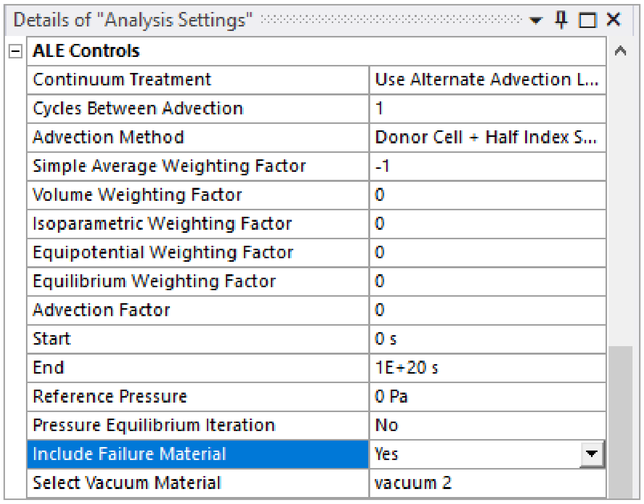
Worksheet

Type	Data Type	Data...	Comp...	Expression	Output Unit
ALE_DENSITY	Element Nodal	Scalar		ALE_DENSITY	Density
ALE_DOMINANT_MATERIAL	Element Nodal	Scalar		ALE_DOMINANT_MATERIAL	No Units
ALE_VOLUME_FRACTION_MAT1	Element Nodal	Scalar		ALE_VOLUME_FRACTION_MAT1	No Units
ALE_SPECIES_MASS_MAT1	Element Nodal	Scalar		ALE_SPECIES_MASS_MAT1	Mass
ALE_VOLUME_FRACTION_MAT2	Element Nodal	Scalar		ALE_VOLUME_FRACTION_MAT2	No Units
ALE_SPECIES_MASS_MAT2	Element Nodal	Scalar		ALE_SPECIES_MASS_MAT2	Mass

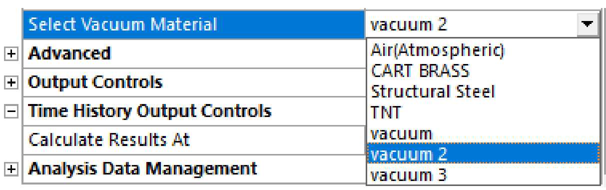
3.7.5.1. ALE\S-ALE Material Failure

Material failure is available in the LS-DYNA system for ALE and S-ALE simulations. This approach replaces the failed material with a dummy vacuum material by creating a dummy part with its own ID and associating a *MAT_VACUUM keyword with the part.

To enable material failure in your simulation, select Yes for the **Include Failure Material** option under the **ALE Controls** category of the LSDYNA system Analysis Settings.

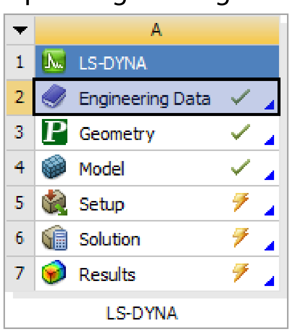


If Yes is selected, a new option called **Select Vacuum Material** appears. Select a vacuum material from the list of materials that displays. The materials in this list come from the materials defined in Engineering Data.

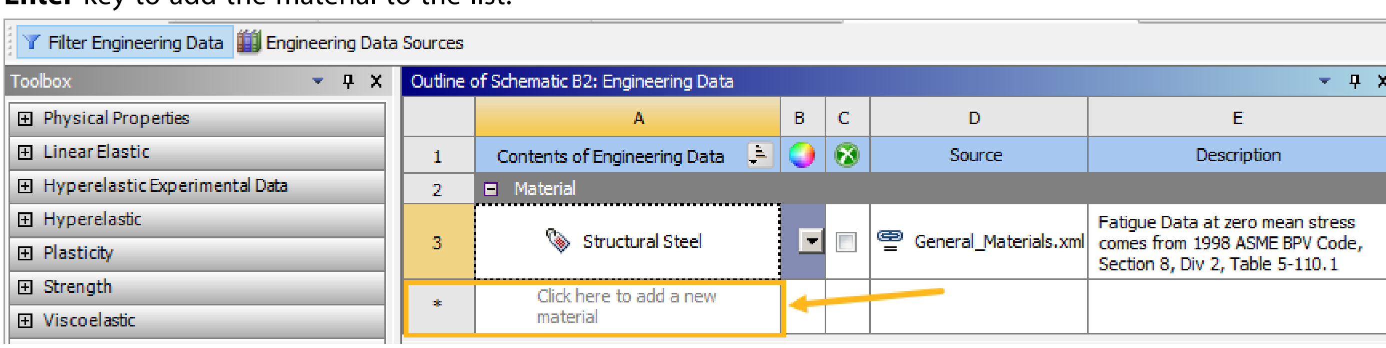


To create a vacuum material:

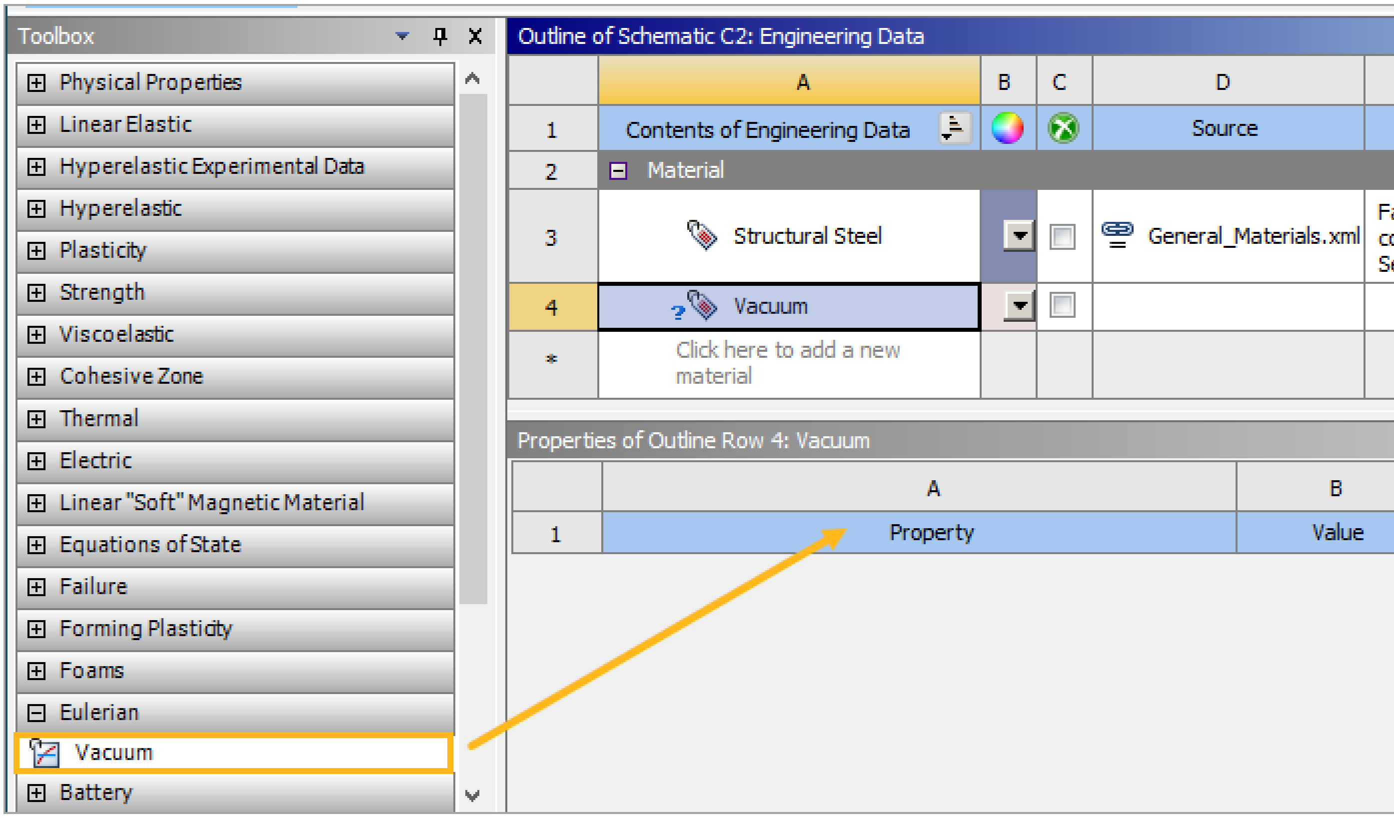
1. Open Engineering Data from the cell in the Project Schematic.



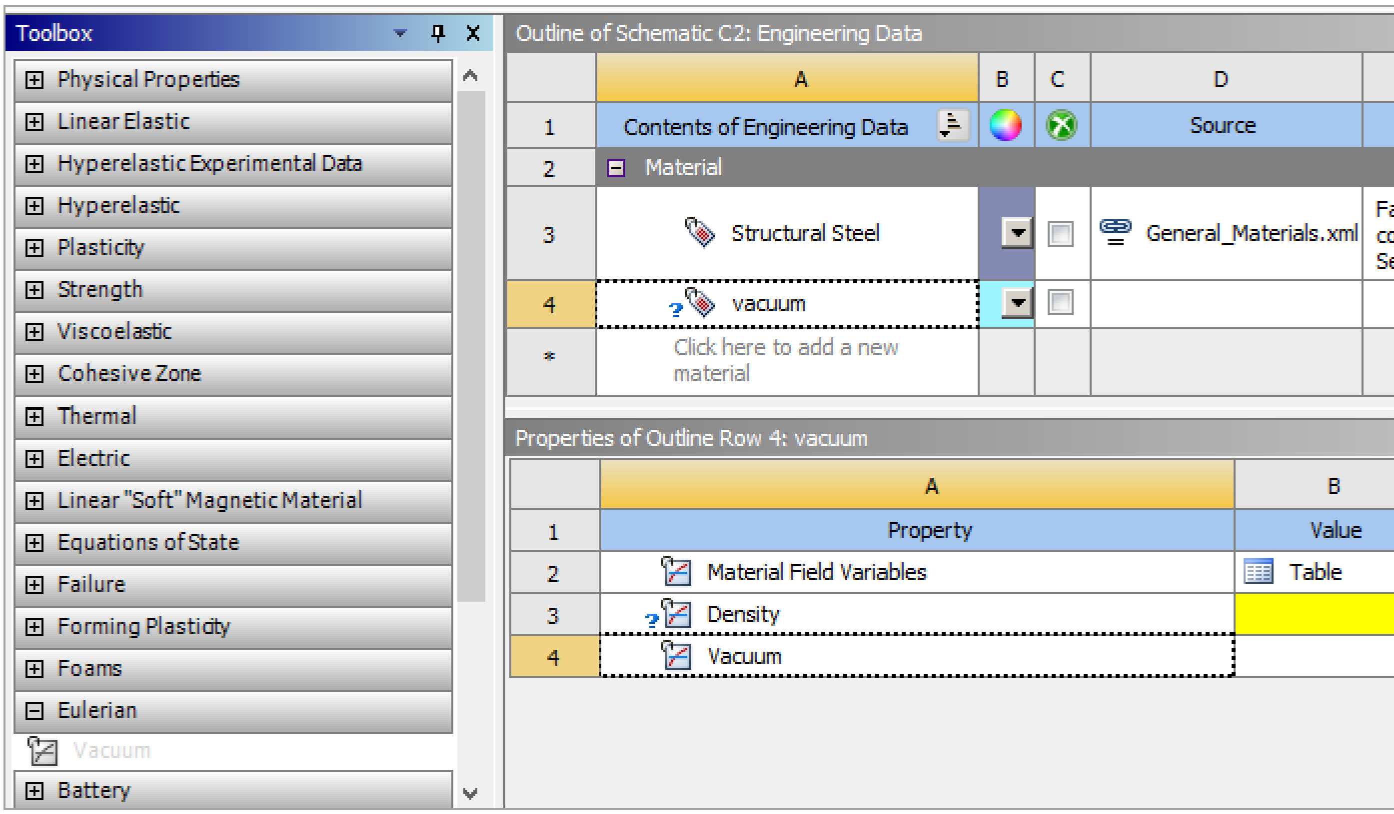
2. Enter the name of the new material where it says **Click here to add new material**. Press the **Enter** key to add the material to the list.



3. In the toolbox on the left-hand side, expand the Eulerian category, right-click Vacuum and select Include Property (or drag Vacuum into the Properties panel).



4. Enter the appropriate Density value to complete the material definition.



The dummy vacuum material will be written in the input .k file as follows:

```
*MAT_VACUUM
$       ID        ro                                                    unused1
        3     1E-19
*PART
$                                                            partname   unused1

$       ID     secid       mid     eosid      hgid      grav     adopt     tmid
        3         3         3         0         0         0         0         0
*SECTION_SOLID
$       ID    elform       aet                                          unused1
        3        11         0
```

Behavior Notes

- If your model contains materials that support failure but you have not set the option **Include Failure Material** to Yes, an error will be reported telling you to do so.
- If the model does not contain materials with failure but you set **Include Failure Material** to Yes, a warning will be issued telling you that the setting is not needed.
- If the model has at least 1 *physical* vacuum part defined, a warning will be issued saying the failed material will convert to the physical vacuum part and not the dummy part that was define by setting the **Include Failure Material** option. If the model has 2 or more physical vacuum parts, there is some ambiguity as to which physical vacuum part will be used to convert the failed material.
- Post-processing of the failed material is best done in LS-PrePost. The Mechanical application has issues showing the converted failed material as vacuum.

3.7.6. SPH Workflow

LS-DYNA provides a Smooth Particle Hydrodynamic (SPH) solver. The state of the system is represented by a set of particles which possesses material property and interact with each other within the range controlled by a weight function. Particle approximation is a truly mesh-free explicit Lagrangian method.

The SPH solver is a good choice in the following types of analyses:

- Large material distortion—for example, crashworthiness, hyper-velocity impact
- Moving boundaries, free surface—for example, fluid and structure interaction
- Adaptive procedure—for example, forging and extrusion

To use this solver, your project must use a Reference Frame of type Particle and include a Particle Method mesh object.

For additional information about setting up an SPH analysis, see the following topics:

- SPH analysis settings (p. 139)
- Adaptive Solid to SPH (p. 19)

- SPH to SPH contact (p. 29)
- Bounding Box (p. 60)

Note:

- The default contact thickness between SPH parts and Lagrangian parts is zero. In some situations, it's recommended that you define a contact thickness in order to detect and avoid penetration between an SPH part and a Lagrangian part. If you set **Contact Thickness** to Yes in the SPH Controls section of the Analysis Settings, the solver will automatically define a contact thickness.
- The SPH part viscosity can be set by scoping the SPH part to an Hourglass Control object and entering the Quadratic and Linear Viscosity. See *Hourglass (p. 159) for more information.

3.7.7. Incompressible SPH Workflow

The Incompressible SPH (ISPH) solver allows you to solve FSI problems that involve structural and fluid bodies. Here is some general guidance for setting up an ISPH analysis using a Mechanical LS-DYNA system.

The fluid part must be a solid body with a Reference Frame of type Particle and you must mesh it with the Particle Method mesh method. To set the material of the fluid part you must scope an ISPH Region (p. 27) object to the body, set the **Type** to fluid, and enter the material properties of the fluid.

If a solid structural body is used in the model, you must extract its skin (using a modeling program like Discovery Modeling) and define it as a new rigid shell body. The shell body should be meshed with triangles, then an ISPH Region should be scoped to the shell body. The interaction surface of the shell body is meshed with SPH particles using the ISPH Region. The ISPH Region will discretize the surface to SPH particles within the solver.

The interaction between the structure and the fluid is handled by the particle formulation of ISPH (particles can't overlap). No contact object is needed since the contact will be handled by the SPH particle formulation. The interaction occurs between the particle meshing on the surface of the structure (shell) body and the particles of the fluid region (solid body).

The Result worksheet shows results for SPH particles that are created inside the solver. See Post-processing for Solver-Created SPH Elements (p. 66).

ISPH is suitable for slower fluid-flow simulations such as automotive water wading and gear box lubrication.

Note:

ISPH is only available for an MPP simulation. You must set the processing type to MPP in the **CPU and Memory Management** section of the Analysis Settings Details panel in order to use it.

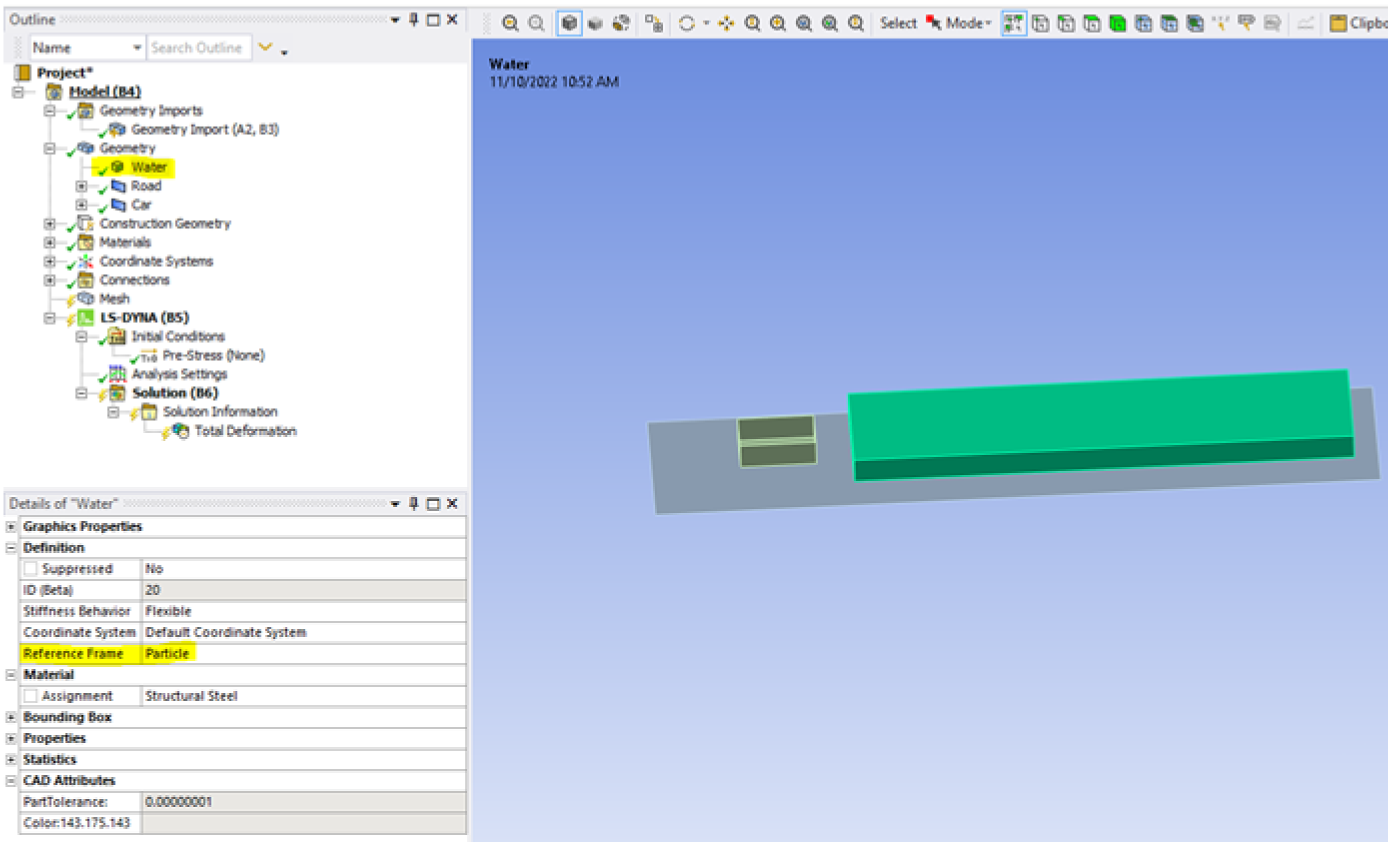
For additional information about setting up an SPH analysis, see the following topics:

- SPH analysis settings (p. 139)
- Bounding Box (p. 60)

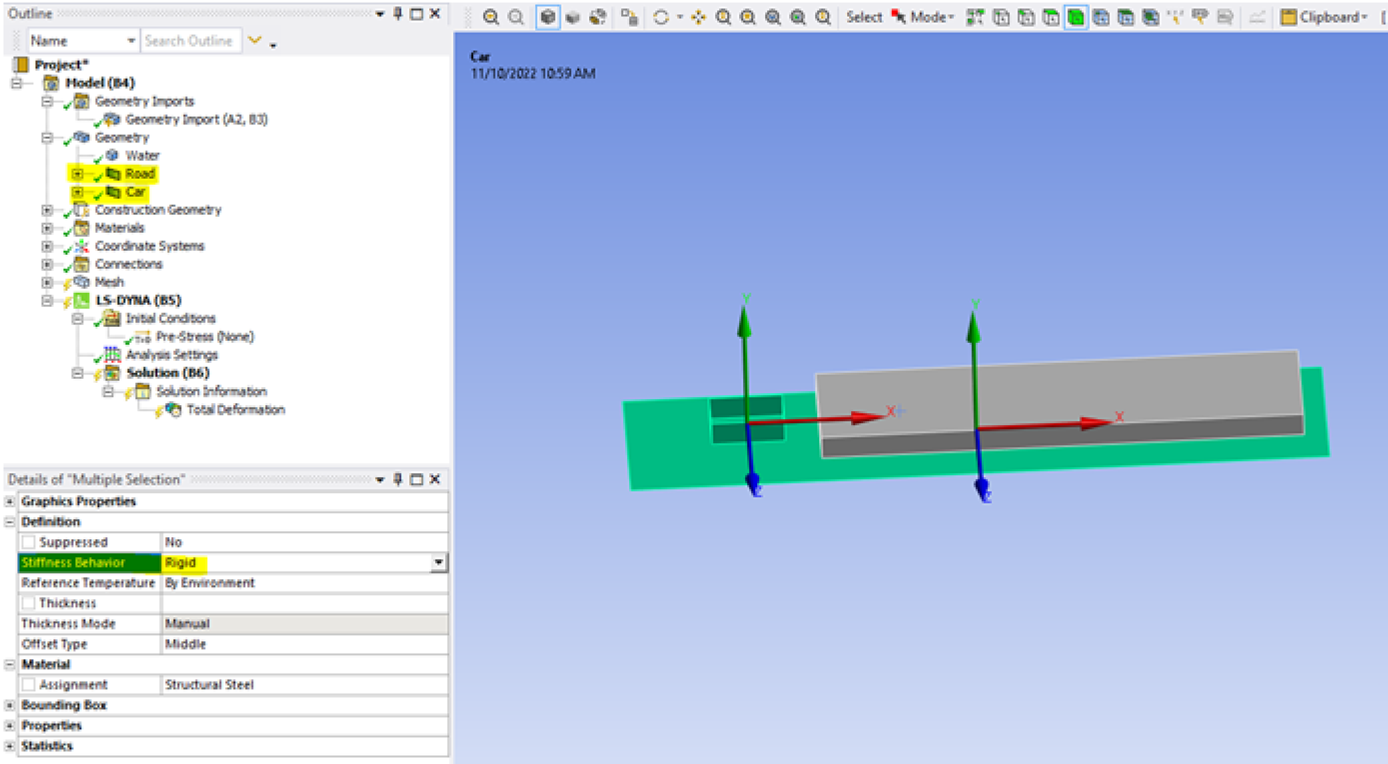
3.7.7.1. An Example of an ISPH Analysis

The following is an example of how to use the ISPH feature in the Workbench LS-DYNA system. The screen shots show an example of a car traveling through a puddle, but the steps describe the general procedure that you can use for any ISPH analysis.

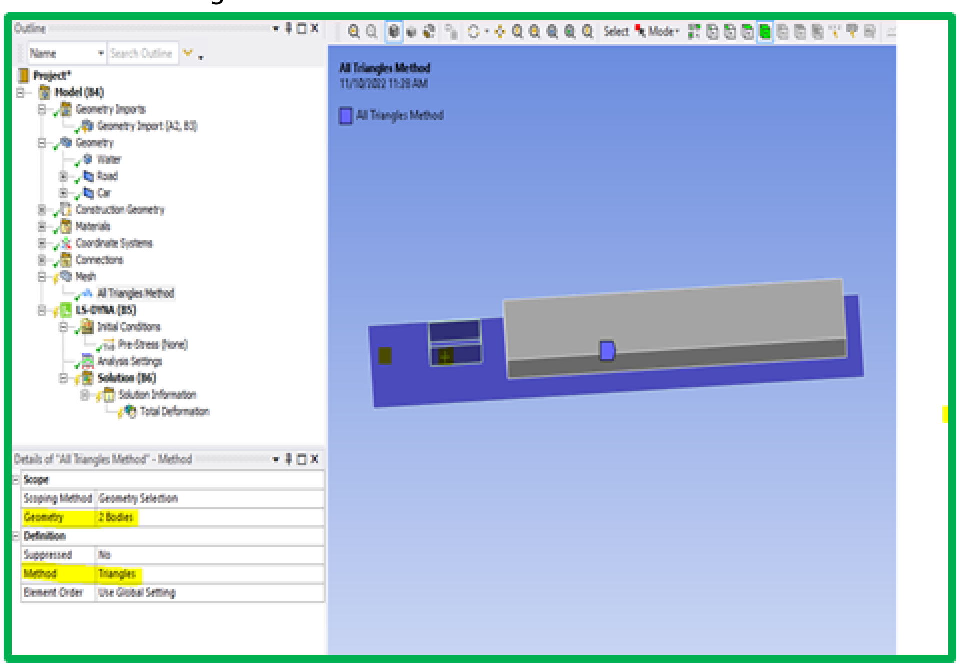
1. Import an external CAD file containing the model into the LS-DYNA system.
2. Set the **Reference Frame** of the solid body that represents the SPH body to Particle.



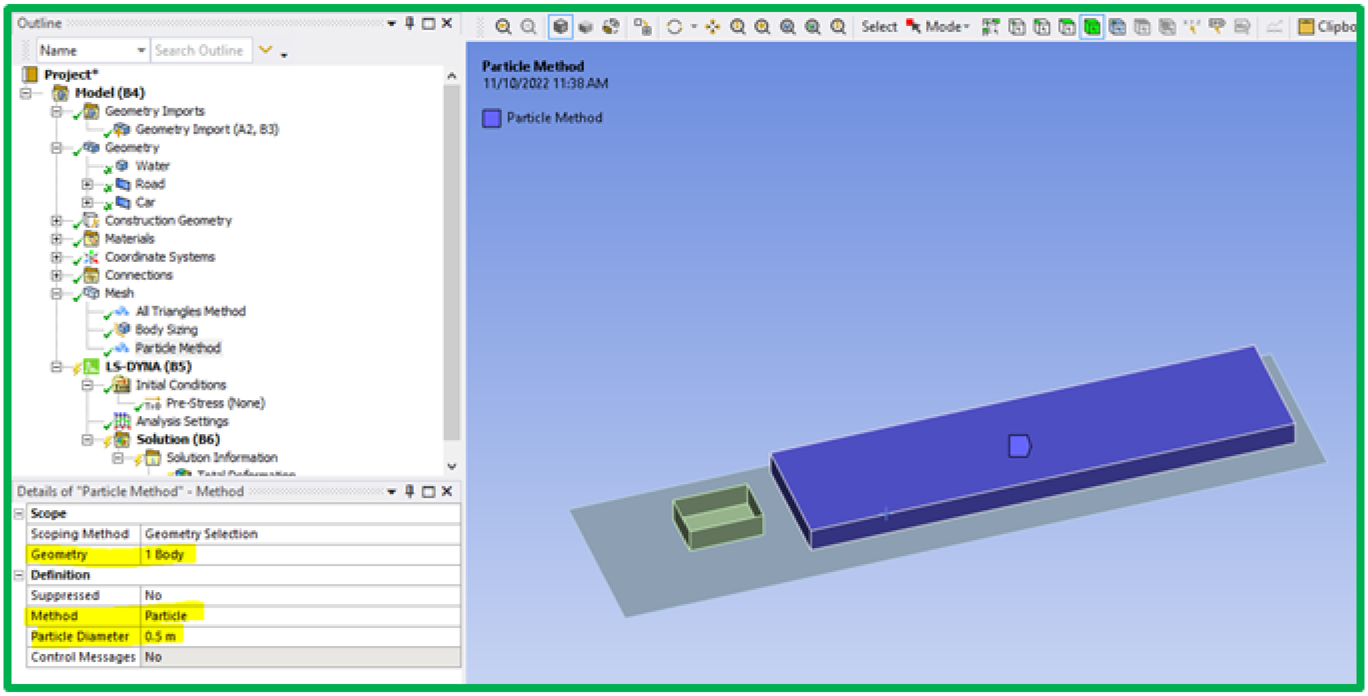
3. Set the **Stiffness Behavior** of the ISPH surface bodies to rigid.



4. Insert a **Mesh Method** and scope it to the ISPH surface bodies. Set **Method** to Triangles so that you mesh the ISPH surfaces with triangles. Insert a **Body Sizing** object under **Mesh** and set a size for the triangle elements.

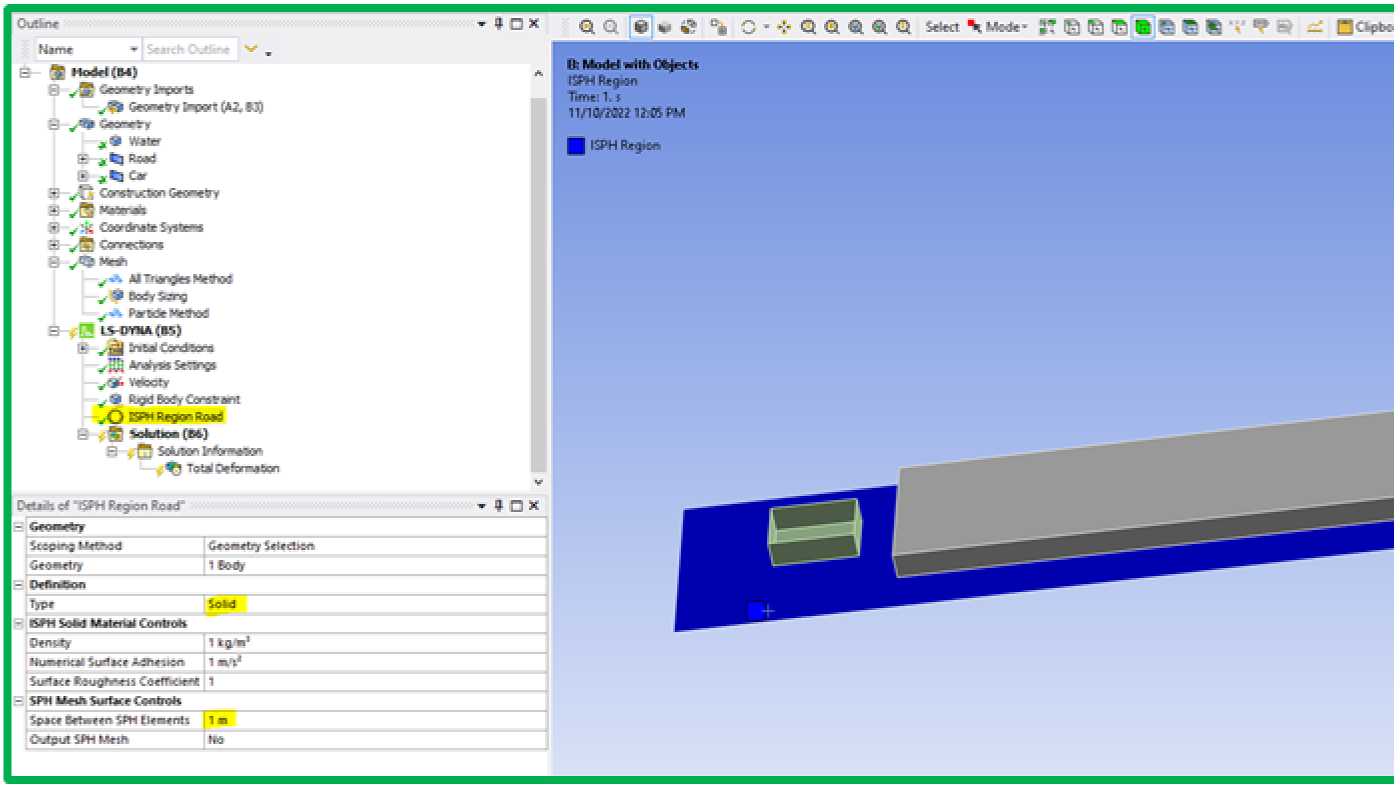


5. Add a **Particle Mesh** method and scope it to the SPH solid body. You can set the diameter of the particles.

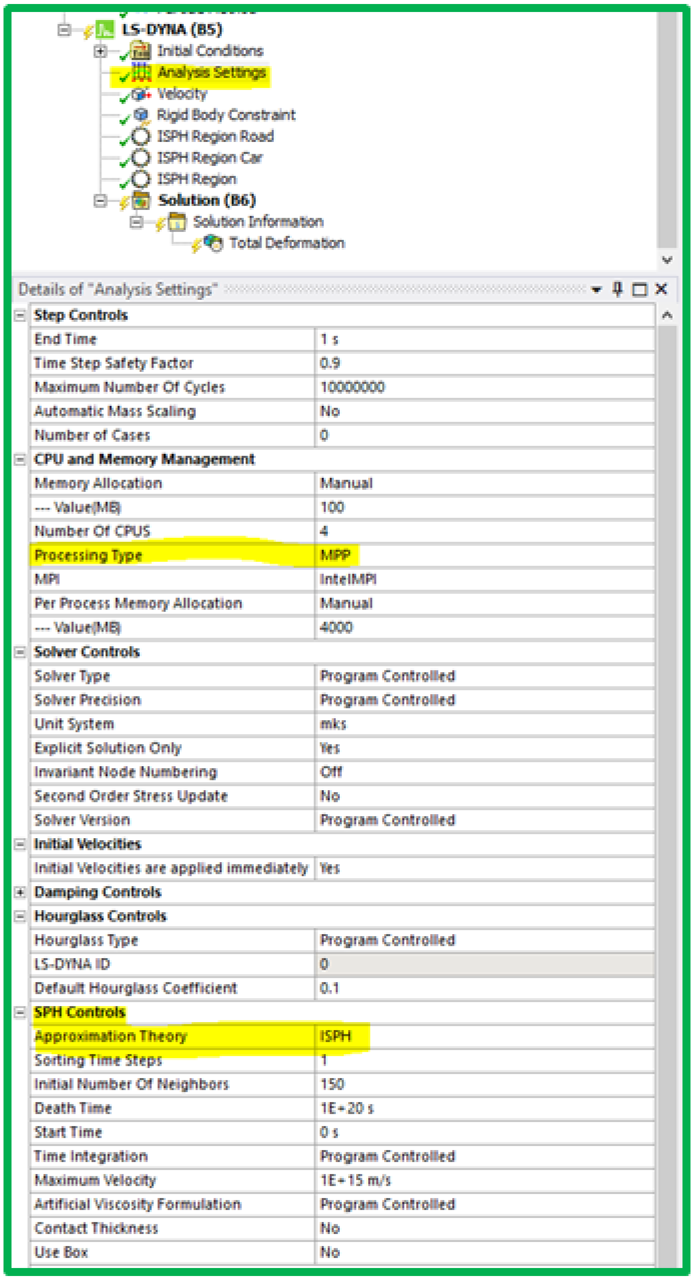


6. Insert any necessary boundary conditions.

7. Insert an ISPH Region for each of the bodies in the analysis. Scope one region to each body. Set the **Type** (physics type) to Solid (for the shell bodies) or Fluid (for the solid SPH body) as appropriate and set the corresponding properties for each Region.



8. In the Analysis Settings properties, under CPU and Memory Management set the **Processing Type** to MPP, and under SPH Controls set the **Approximation Theory** to ISPH.



The setup of the ISPH analysis is complete. Add any additional objects that you need and you are ready to solve the model.

3.7.8. SPG and ISPG Workflows

LS-DYNA provides two types of Smoothed Particle Galerkin (SPG) solvers:

- The standard SPG solver used in the simulation of destructive manufacturing process such as riveting, screwing, drilling, and machining.
- The Incompressible SPG (ISPG) Solver used for the simulation of incompressible free surface fluid flow such as the reflow process of solder joints in electronic equipment.

SPG and ISPG are mesh-free methods that use a finite element mesh, unlike SPH which requires the generation of a specific particle mesh. The SPG and ISPG bodies must be solid.

To apply SPG or ISPG methods on a part, scope the part in a section object with the method property set to SPG or ISPG. Additional information on SPG and ISPG are described in the following items:

- *SECTION_SOLID_SPG (p. 156)
- *SECTION_SOLID_FPD, *MAT_IFPD (p. 158)

To handle the interaction between a finite element part and an ISPG part, there is a new coupling object called **ISPG to Surface Coupling**. For additional information on this object, see:

- ISPG to Surface Coupling (p. 30)

To use the ISPG solver, the Solver Precision in the Analysis Settings must be Double and the mesh for the ISPG body must use tetrahedral elements. If these two conditions are not met, error messages that indicate the problem will be generated.

When you are using MPP for an SPG analysis, the card *CONTROL_MPP_DECOMPOSITION_PARTS_DISTRIBUTE is written to the input file. This card specifies the ID of a *SET_PART_LIST that includes all SPG part IDs. For further details, see *CONTROL_MPP_DECOMPOSITION_PARTS_DISTRIBUTE in the LS-DYNA Keyword and Theory Manuals.

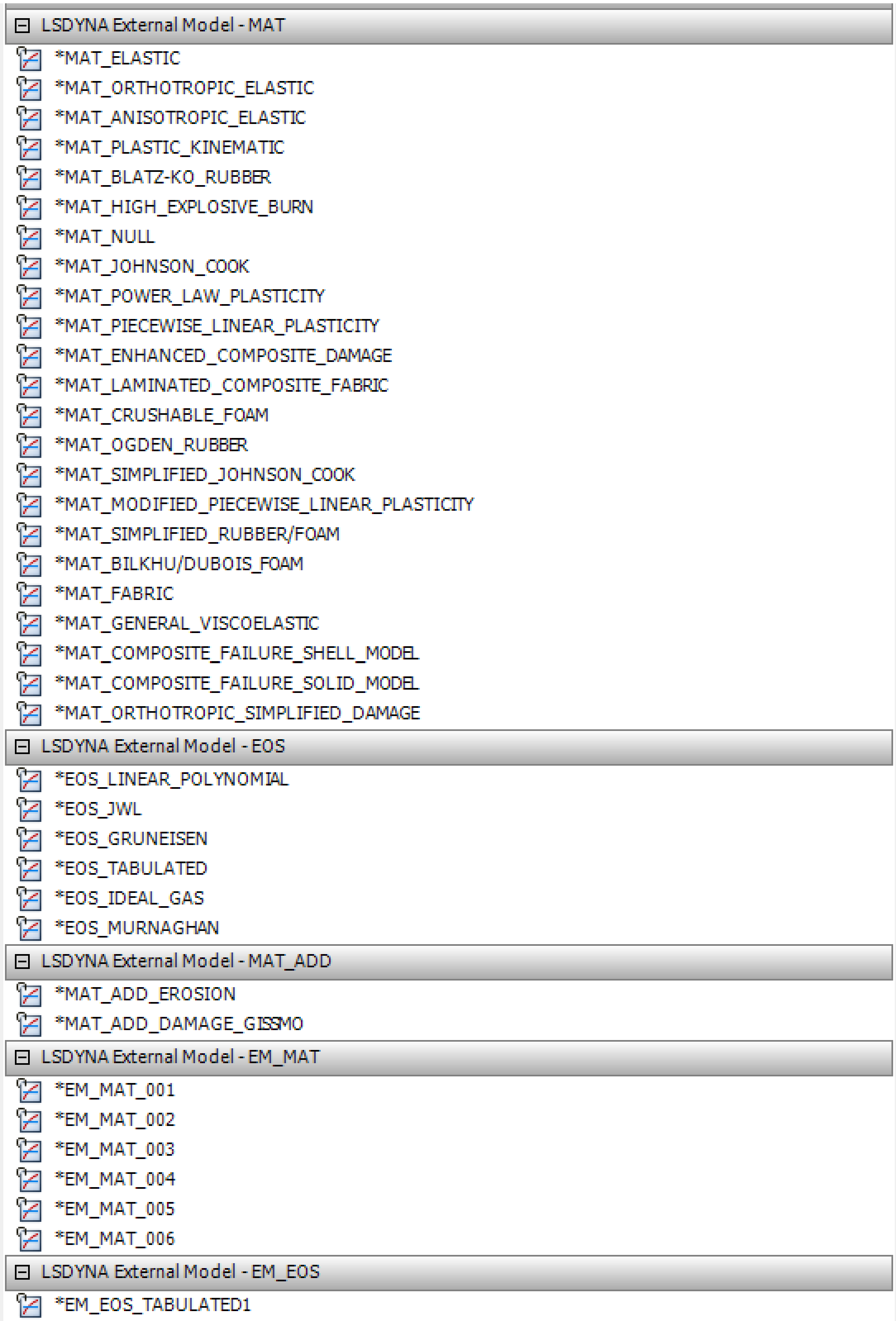
3.7.9. Electromagnetic Workflow

When the **Solver Type** field in Solver Controls category of the Analysis Settings object is set to Coupled Structural Thermal Electromagnetic Analysis, the LS-DYNA system in Mechanical lets you perform an analysis using electromagnetic boundary conditions to evaluate the resistive heating effects in the model. When this solver type is selected, the Analysis Settings object provides **EM Solver Controls** (p. 213) and **EM Step Controls** (p. 214) categories with additional options.

Note:

- The only EM solver type currently available is Resistive Heating.
- The only unit system currently available for the EM workflow is MKS.

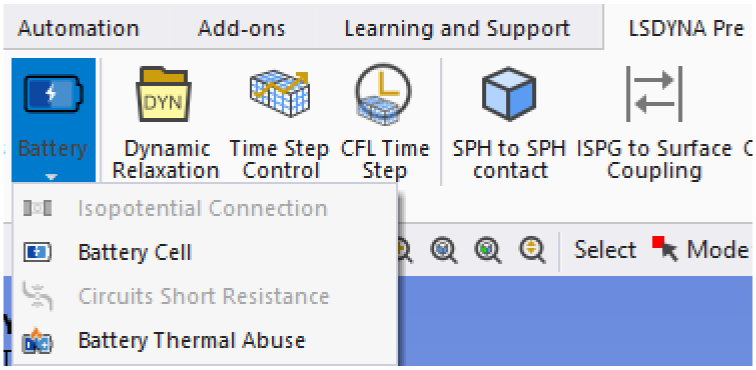
Electromagnetic material properties can be added to materials from Engineering Data in Workbench. The native electric properties supported are: Isotropic Resistivity, Orthotropic Resistivity, and Isotropic Relative Permeability. In Engineering Data, you can also select LS-DYNA cards to add electromagnetic properties to materials. The cards are EM_MAT_001, EM_MAT_002, EM_MAT_003, EM_MAT_004, EM_MAT_005, EM_MAT_006, and EM_EOS_TABULATED1.



The LS-DYNA system provides support for the following Mechanical electromagnetic loads: Voltage and Current. These loads are implemented with the keyword *EM_BOUNDARY_ PRESCRIBED.

3.7.9.1. Battery Support

There are several objects that can be defined in an LS-DYNA system that provide support for modeling batteries.



These can be found in the Battery section of the LS-Pre tab and include:

3.7.9.1.1. Battery Cell

A Battery Cell object allows you to model a battery cell using the distributed circuit parameters for Randles Batmac Model. The Batmac Model is a macro battery model where solid elements are retained for the solid mechanics and thermal solve and where each conducting node will have its own Randles circuit associated to it.

Add a Battery Cell object under your LS-DYNA environment in the Project tree. You can then define the following properties for the Battery Cell:

Randles Circuit Type

The Randles Circuit Type defines the order of the equivalent circuit model.

A Randles circuit consists of an ideal voltage source, an internal resistance and n parallel RC circuits, n being the order of the model. A Randles circuit with only one RC loop is said to be 1^{st} order.

Initial State of Charge

The Initial State of Charge defines the initial state of charge of the Cell.

Voltage

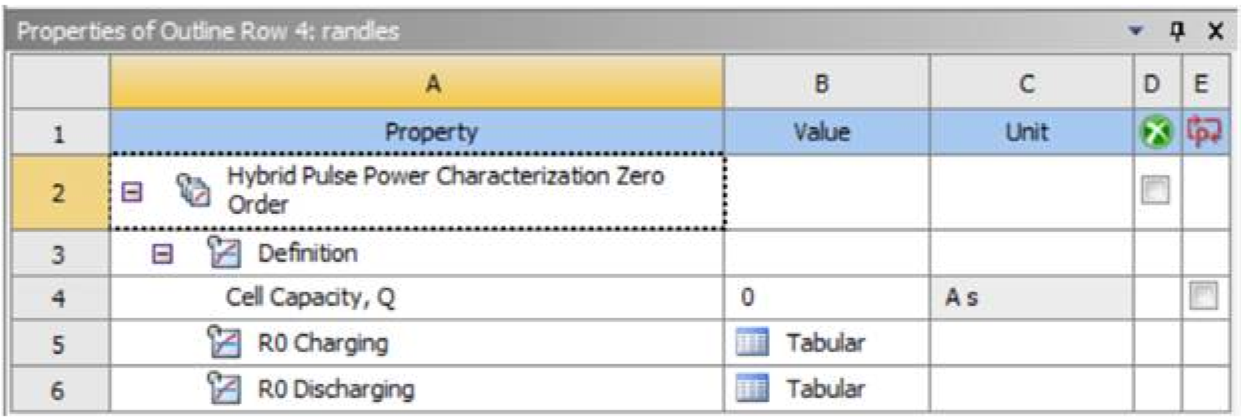
Voltage represents the equilibrium voltage (OCV) as function of the state of charge.

Hybrid Pulse Power Characterization

Hybrid Pulse Power Characterization (HPPC) is a testing method that evaluates the performance of cell Batteries. It is typically the voltage measurement of a cell following a series of 10s charge and discharge pulses at different state of charges (SOCs) at a given temperature.

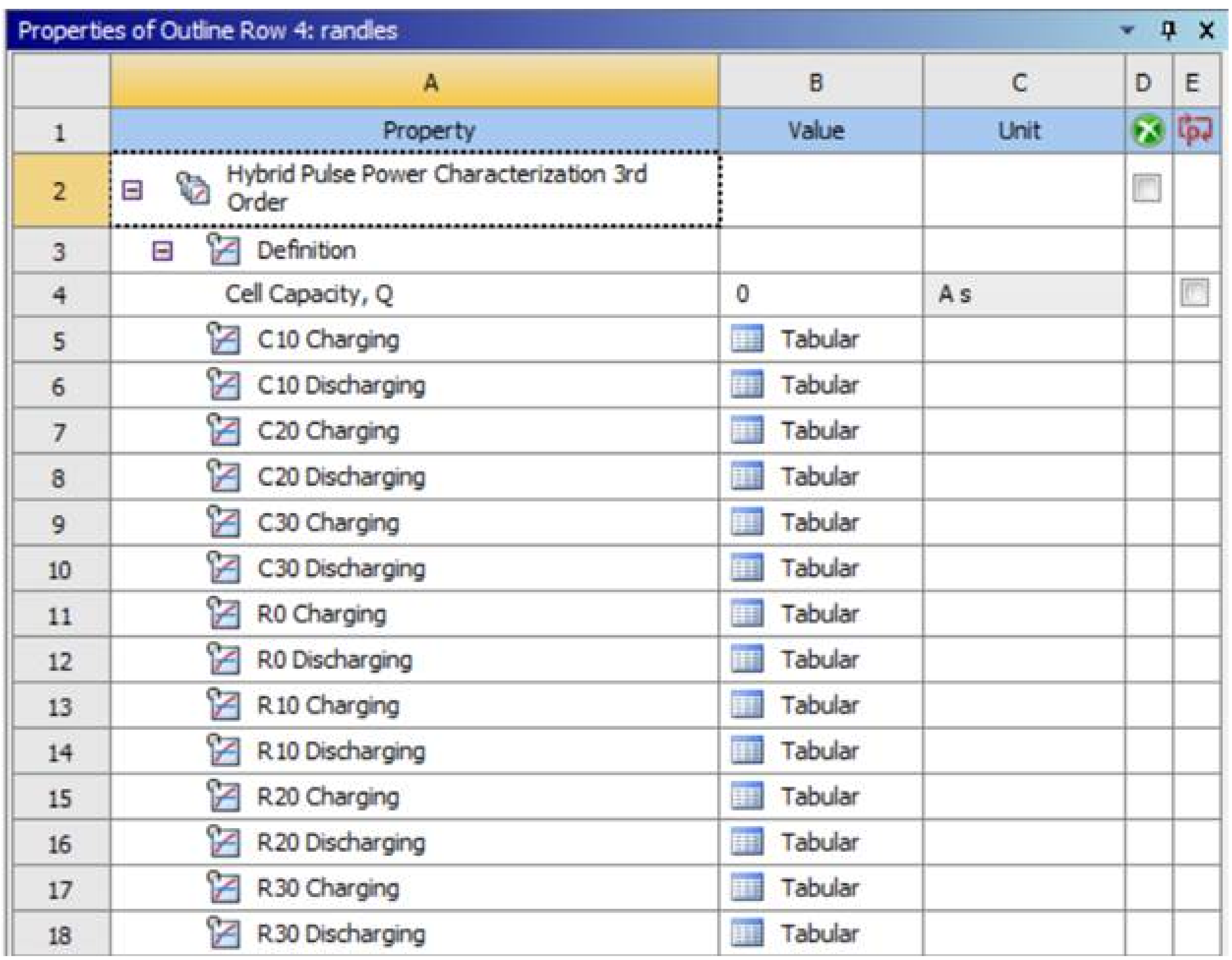
The resistance and capacitance parameters are determined through identification from those measurements. They represent resistance or capacitance as function of the state of charge, or resistance or capacitance as function of state of charge and the temperature.

The HPPC characterization is available in Engineering Data for Zero Order, First Order, Second Order and Third Order Randles Circuits.



Properties of Outline Row 4: randles

	A	B	C	D	E
1	Property	Value	Unit		
2	Hybrid Pulse Power Characterization Zero Order				
3	Definition				
4	Cell Capacity, Q	0	A s		
5	R0 Charging	Tabular			
6	R0 Discharging	Tabular			



Properties of Outline Row 4: randles

	A	B	C	D	E
1	Property	Value	Unit		
2	Hybrid Pulse Power Characterization 3rd Order				
3	Definition				
4	Cell Capacity, Q	0	A s		
5	C10 Charging	Tabular			
6	C10 Discharging	Tabular			
7	C20 Charging	Tabular			
8	C20 Discharging	Tabular			
9	C30 Charging	Tabular			
10	C30 Discharging	Tabular			
11	R0 Charging	Tabular			
12	R0 Discharging	Tabular			
13	R10 Charging	Tabular			
14	R10 Discharging	Tabular			
15	R20 Charging	Tabular			
16	R20 Discharging	Tabular			
17	R30 Charging	Tabular			
18	R30 Discharging	Tabular			

3.7.9.1.2. Battery Thermal Abuse

Safety has become an important issue in battery design. Under abuse conditions, thermal runaway may occur in a battery. Increased temperature could trigger thermal runaway reactions. Excessive of heat released as a result of such conditions could damage a battery cell, or even cause fire or explosion. Thermal runaway reactions are very complicated and material-specific.

Thermal Abuse is specified on a part of the Geometry.

For modeling thermal runaway reactions and simulating a battery thermal behavior under thermal abuses, the LS-DYNA system offers two semi-empirical models:

- One-Equation Model: In the one-equation model, thermal runaway reactions are lumped together as one reaction.
- Four-Equation Model: In the four-equation model, thermal runaway reactions are put into the following four categories:
  - solid electrolyte interface (SEI) decomposition reactions
  - negative electrode-electrolyte reactions
  - positive electrode-electrolyte reactions
  - electrolyte decomposition reactions

These models are available in Engineering Data under the Battery Folder. You need to create a material model and add the properties from one of these models to that material model.

To simulate thermal Abuse, add a Battery Thermal Abuse object under your LS-DYNA environment in the Project tree. You can then select the material you want to use from the **Thermal Abuse** dropdown.

Define the following properites for the Thermal Abuse:

Birth Time

Birth Time for application of heat source term.

Death Time

Death Time for application of heat source term.

Minimum Temperature

Minimum temperature before heat source activation is triggered.

Maximum Temperature

Maximum temperature before heat source activation is triggered.

3.7.9.1.3. Isopotential Connections

An Isopotential Connection defines a connection between two isopotentials. An Isopotential constrains nodes of a selection so they have the same potential value. The isopotential connection object inserts a specified circuit component between two isopotentials.

The circuit component can be one the following types:

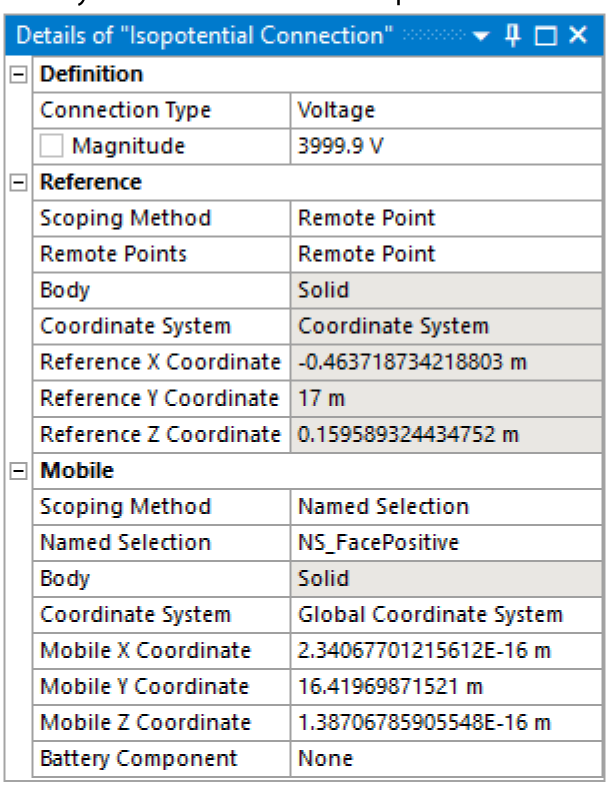
- Short circuit
- Resistance
- Voltage
- Current
- RLC circuit

Applying an Isopotential Connection

To apply an Isopotential Connection:

1. After importing the model, highlight the **Model** object in the tree and choose the **Connections** option from the Context tab.
2. Highlight the new **Connections** object and select Isopotential Connection from the Context tab.

Once you have inserted the Isopotential Connection, define its properties.



Connection Type

The **Connection Type** can be one of the following types :

- Short circuit
- Resistance
- Voltage
- Current
- RLC circuit

Resistance, Voltage (Source), or Current (Source) can be a constant value, or have values which vary over time. In the latter case, select Resistance over Time, Voltage over Time, or Current Over Time as the Connection Type.

Scoping

Define the Isopotential Connection’s end points using the properties in the Reference and Mobile categories. If one end is intended to be Ground, you must set the **Scoping Method** for the Reference to Ground. In this case, model scoping is only available on the Mobile side. Since the Isopotential Connection is unidirectional, these two locations determine the Isopotential Connection's line of action. As such, the Isopotential Connection's reference and mobile locations cannot be the same as this would result in a Isopotential Connection with zero length.

The reference and mobile locations are used to graphically display the connection, and each is initialized at the barycenter of the corresponding selections.

You can scope an Isopotential Connection to a:

- Single body or to multiple bodies.
- Single face or to multiple faces.
- Single edge or multiple edges.
- Single vertex or multiple vertices.

Note:

Note: Connections are classified as remote boundary conditions.

Battery Component

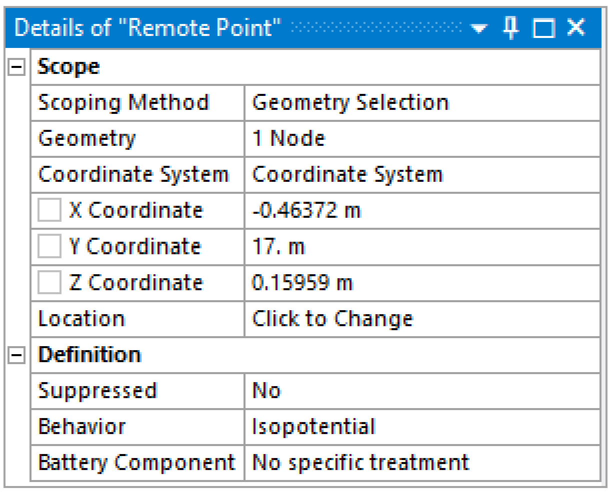
You can define each Reference and Mobile isopotential to behave as a **Battery Component**. The following options are available:

- None
- Current Collector Positive

- Positive Electrode
- Separator
- Negative Electrode
- Current Collector Negative

Remote Point Definition

If the **Scope Method** property of the Isopotential Connection is set to Remote Point, the Isopotential Connection will then assume the behavior defined in the referenced Remote Point as well as other related properties. The remote point **Behavior** must be set to Isopotential.


Details of "Remote Point"

Scope	
Scoping Method	Geometry Selection
Geometry	1 Node
Coordinate System	Coordinate System
X Coordinate	-0.46372 m
Y Coordinate	17. m
Z Coordinate	0.15959 m
Location	Click to Change
Definition	
Suppressed	No
Behavior	Isopotential
Battery Component	No specific treatment

The Remote Point can be defined to behave as a battery component by setting **Battery Component** to one of the following:

- None
- Current Collector Positive
- Positive Electrode
- Separator
- Negative Electrode
- Current Collector Negative

In cases where the Remote Point is attached to a material that has EM_MAT_001 as a property, the value Application in EM_MAT_001 must match the value of Battery Component.

3.7.9.1.4. Circuits Short Resistance

Internal short circuits may occur in a cell battery because of structural deformation due to the stress happening during a crash, or some other mechanical event. The Circuits Short Resistance object sets the value of the circuit resistance when a short occurs, and specifies conditions that cause a short to occur. There can be only one Circuits Short Resistance item in an analysis and the conditions it defines are used in all circuit elements to determine if a short occurs.

Condition

Short circuits can occur if a Stress Limit, a Temperature condition, or Strain condition are met. These conditions can trigger the short circuit independently of each other, or if a group of some of them occur at the same time. If the condition is:

- **Stress Limit**: The short circuit happens if the stress in the cell is higher than the Stress Limit
- **Maximum Temperature**: The short happens if the temperature in the cell is higher than the Maximum Temperature
- **Maximum Strain**: The short happens if the strain in the cell is higher than the Maximum Strain

Stress Limit __ Maximum Temperature

Defines the conditions based on the temperature and the stress in the model. The Operator (And or Or) indicates if the two conditions are dependent or not.

Stress Limit __ Maximum Strain

Defines the conditions based on the strain and the stress in the model. The Operator (And or Or) indicates if the two conditions are dependent or not.

Maximum Temperature __ Maximum Strain

Defines the conditions based on the strain and the temperature in the model. The Operator (And or Or) indicates if the two conditions are dependent or not.

Stress Limit __ Maximum Temperature __ Maximum Strain

Defines the conditions based on the strain, the stress, and the temperature in the model. The two Operators (And or Or) indicate if the three conditions are dependent or not.

Resistance

Value of the Resistance in Ohms.

Von Mises

Von Mises Stress value which triggers the short circuit.

Temperature

Temperature value which triggers the short circuit.

Effective Strain

Effective Strain value which triggers the short circuit.

3.7.10. Composites Workflow

Composites can be added either through the Layered Section object of the Mechanical application, or by importing data from an ACP-Pre system through the Imported Plies object. The **Composite Controls** section of the Analysis Settings provides several options for the method used to define composites in the LS-DYNA input file.

Composite Controls	
Shell Layered Composite Damage Model	Enhanced Composite Damage
Composite Representation	Element

Shell Layered Composite Damage Model

The **Shell Layered Composite Damage Model** option allows you to pick the keywords used to define composite damage models in the LS-DYNA input file. If the damage model is set to Enhanced Composite Damage, the composite plies will be defined using the *MAT_ENHANCED_COMPOSITE_DAMAGE keyword. If the damage model is set to Laminated Composite Fabric, the composite plies will be defined using *MAT_LAMINATED_COMPOSITE_FABRIC.

Composite Representation

The **Composite Representation** option allows you to pick the keywords used to define composites in the LS-DYNA input file. If the representation is set to Element, the composite plies will be defined using the ELEMENT_COMPOSITE family of keywords. If the representation is set to Part, the composite plies will be defined using the PART_COMPOSITE family of keywords.

Note:

In a simulation involving thermal coupling, Part is the only applicable option and will be the one used independent of the **Composite Representation** setting.

3.7.11. Acoustics Workflow

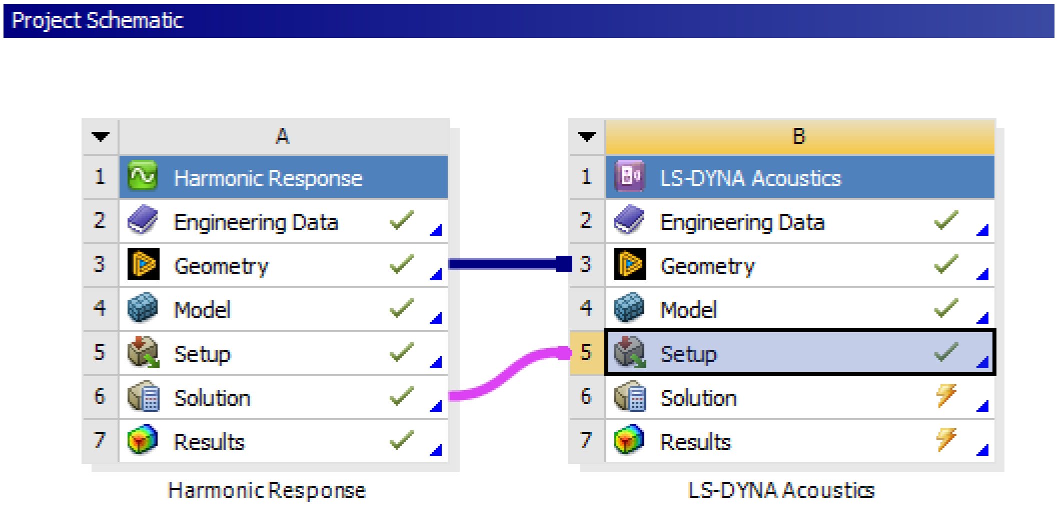
The LS-DYNA BEM (Boundary Element Method) Acoustics analysis is used to determine the frequency response of a structure and the surrounding fluid medium to loads and excitations that vary sinusoidally (harmonically) with time. However, the fluid region in the BEM method is not meshed. The BEM is a numerical computational method of solving linear partial differential equations which have been formulated as integral equations (in boundary integral form).

Since the LS-DYNA Acoustics system shares much of its functionality with the Mechanical Harmonic Response system, the system is primarily documented in LS-DYNA Acoustics Analysis in the *Mechanical Acoustic Analysis Guide*. Included here is a discussion of how to import a velocity into an LS-DYNA Acoustics analysis because currently it cannot be done directly.

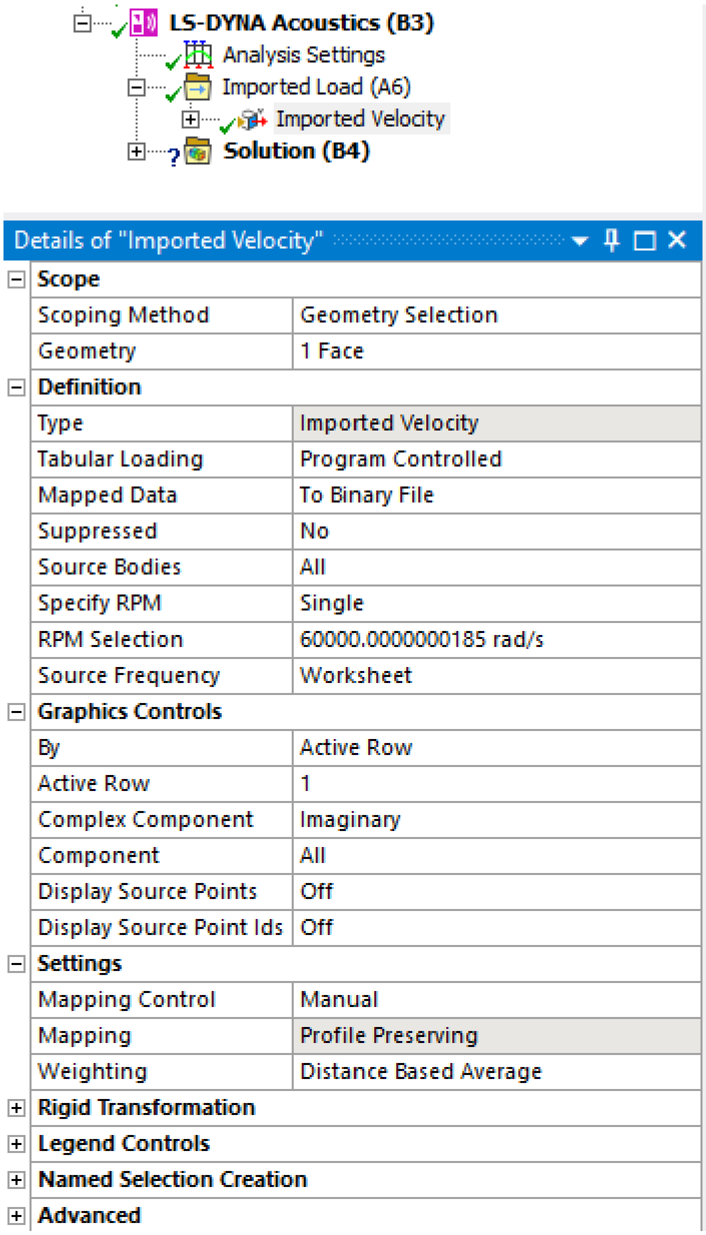
Velocity Import

To import velocities into an LS-DYNA Acoustics system you must use a Harmonic Response analysis as described here.

In the Workbench Project Schematic, link a Harmonic Response system to an LS-DYNA Acoustics system.



The application automatically inserts an **Imported Velocity** object into the downstream system.



See Applying Imported Boundary Conditions in the *Mechanical User's Guide* for more information. The **Imported Velocity** uses the new Load Mapping Workflow Specification. For more information regarding the properties of the **Imported Velocity** object, see Imported Load (Group) in the *Mechanical Object Reference*.

The **Analysis Settings** options such as **RPM Value**, **Step Frequency Range Minimum**, **Step Frequency Range Maximum**, and **Step Solution Intervals** are set based on the **Specify RPM, RPM Selection**, and **Source Frequency** values in the **Imported Velocity** object.

3.7.12. Modifying Default Solver Settings

The LS-DYNA Workbench system uses default solver settings for some LS-DYNA Control Cards (keywords starting with *CONTROL) and for some settings affecting the results file content (keywords starting with *DATABASE).

These settings are not available in the user interface, nor are they written when the corresponding option is set to Program Controlled. If you wish to use different default settings, set the option **Default Solver Control Cards** located under Advanced in the Analysis Settings Details panel to Omit, which will omit these default solver cards. By default, the option is set to Keep.

*CONTROL_TERMINATION and *DATABASE_BINARY_D3PROP are the only cards written when Omit is selected. You can provide your preferred settings by using Keyword Snippets (p. 54).

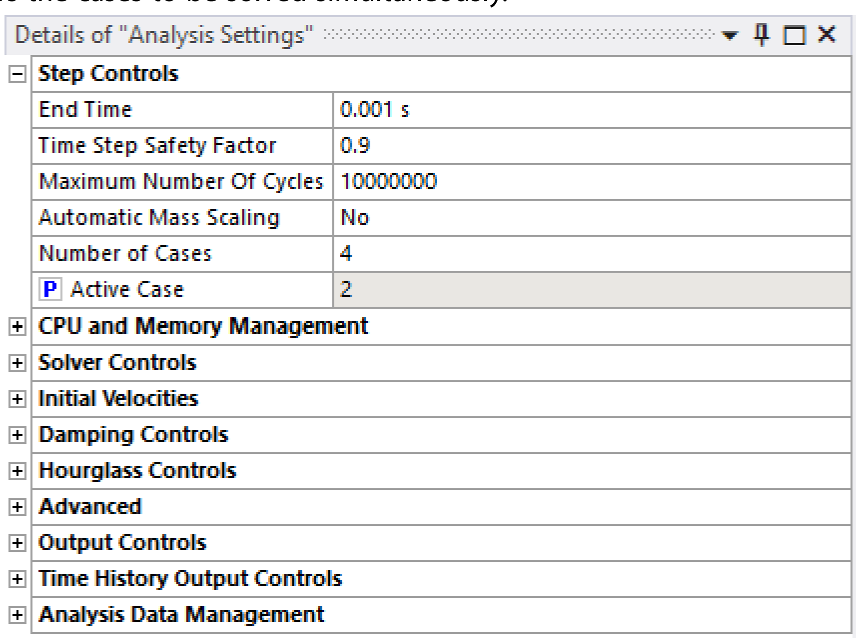
3.7.13. Multiple Case Workflow

The *CASE command enables running multiple LS-DYNA analyses or cases sequentially by submitting a single input file. For further details, see *CASE in the LS-DYNA Keyword User's Manual. The LS-DYNA system includes several options to support the setup of a multiple case workflow.

3.7.13.1. Specifying Cases

The **Number of Cases** property is set under the **Step Controls** section of **Analysis Settings**. This property defines the number of cases. When greater than 0, the CASE input argument is passed to the solver to enable initialization of a multiple case solve. The default value is 0, which indicates that the *CASE command is not active.

The **Active Case** property is set under **Step Controls** section of **Analysis Settings**. This property defines the currently active case. The default value is 0, which indicates that all cases should be solved. If a case number between 1 and **Number of Cases** is specified, then only that case will be solved. In this scenario, the **Case Number** property set under the result objects becomes hidden and the result retrieved will be for the **Active Case**. The **Active Case** property is parameterizable. When set, each case is treated as a separate design point which, with the appropriate settings in Workbench, allows the cases to be solved simultaneously.



The **Case Number** can be set under the following objects:

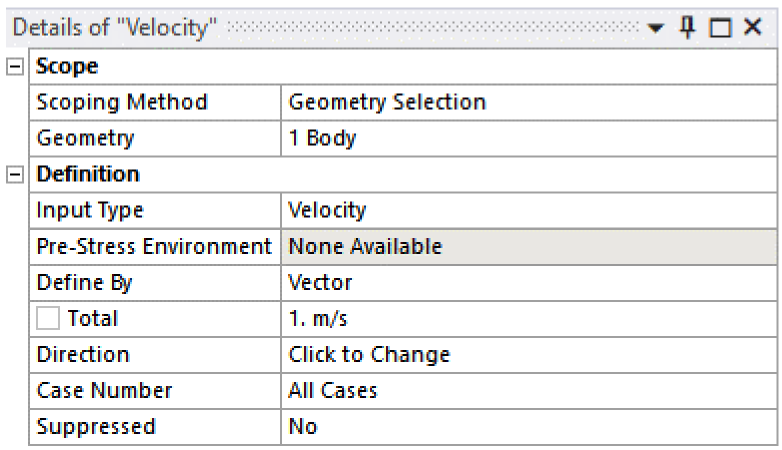
- Initial Conditions (Velocity, Angular Velocity, Drop Height)
- Fixed Support

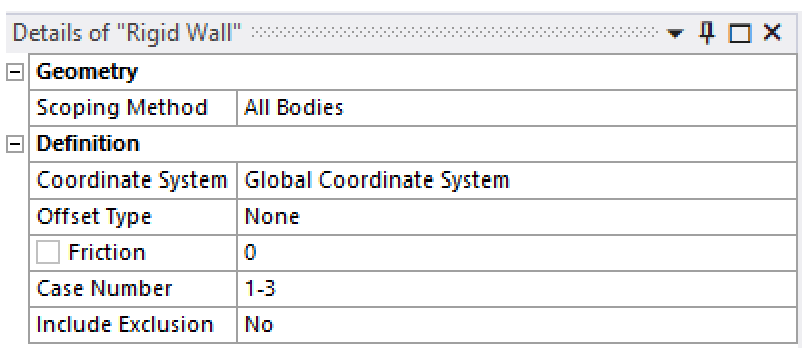
- Displacement
- Velocity
- Rigid Wall

This property provides the option to select a case number between 1 and **Number of Cases** specified under Analysis Settings. The input string accepts a comma separated list of case numbers and/or ranges of case numbers. It should be of the form *a*, *b-c*, *d*, where *a*, *b*, *c*, *d* are integer case numbers, *b-c* represents the range of case numbers from *b* to *c*. For example:

- Case Number = 1-3 (Load applies to case 1, 2 and 3)
- Case Number = 1, 3 (Load applies to case 1 and 3)
- Case Number = 1-2, 4-5 (Load applies to case 1, 2, 4, and 5)

This field indicates the case numbers for which the object is active. In the input file, the keyword corresponding to the object is wrapped in the *CASE_BEGIN_N and *CASE_END_N keywords to indicate to the solver that it must be applied to that specific case. The default value is **All Cases**, which means the keyword corresponding to the object will be applied in all cases.



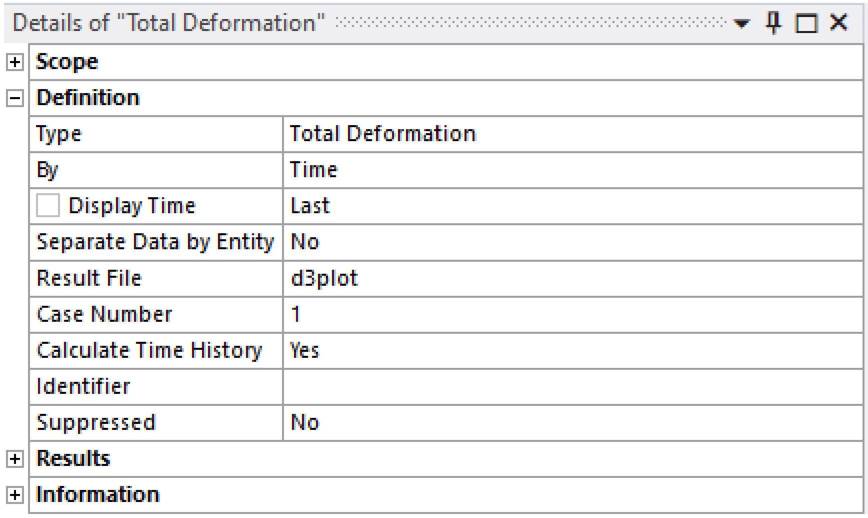


Note:

At least one object (for example, Velocity or Rigid Wall) must be defined for each case from number 1 to **Number of Cases**, otherwise the solver will report an error.

3.7.13.2. Post-processing

When running a multiple case solve, a separate set of result files is created for each case. The **Case Number** property is set under the result object. This property provides the option to select a case number between 1 and **Number of Cases** specified under Analysis Settings. The default value is 1, which indicates that the result shown is for the first case.



Note:

The Worksheet data shown by selecting **Solution Output** under Solution Information is for the first case only.

3.7.13.3. Limitations

Here are the limitations of performing a multiple case workflow:

- Running a Restart Analysis from an initial analysis with multiple cases is not currently supported within the LS-DYNA system.

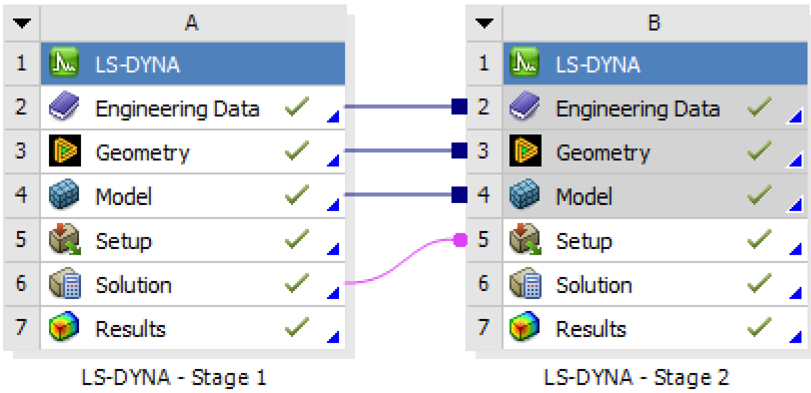
3.7.14. Multi-System Analysis

A multi-system analysis is a set of linked LS-DYNA analyses. Each analysis runs the solver with a regular input file but with some additional initialization using data from an upstream (preceding) LS-DYNA analysis. The data available for initialization to a downstream LS-DYNA analysis includes the deformed geometry positions and stress state. The data used to initialize subsequent analyses is passed to the solver via the binary `dynain.lsda` file.

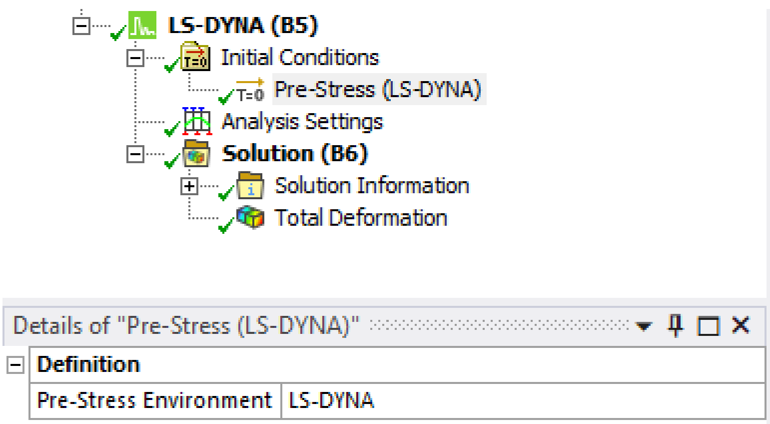
This process has similarities to the full restart described in section Restarting an LS-DYNA Analysis (p. 110) and also the sequential solution described in section Explicit-to-Implicit Sequential Solutions (p. 9). There are two methods for connecting the sequential LS-DYNAj systems, one where the model is shared by both systems, and another where the solved model from the first system becomes the initial model for the second system.

3.7.14.1. Setting up a Multi-System Analysis with Shared Model Data

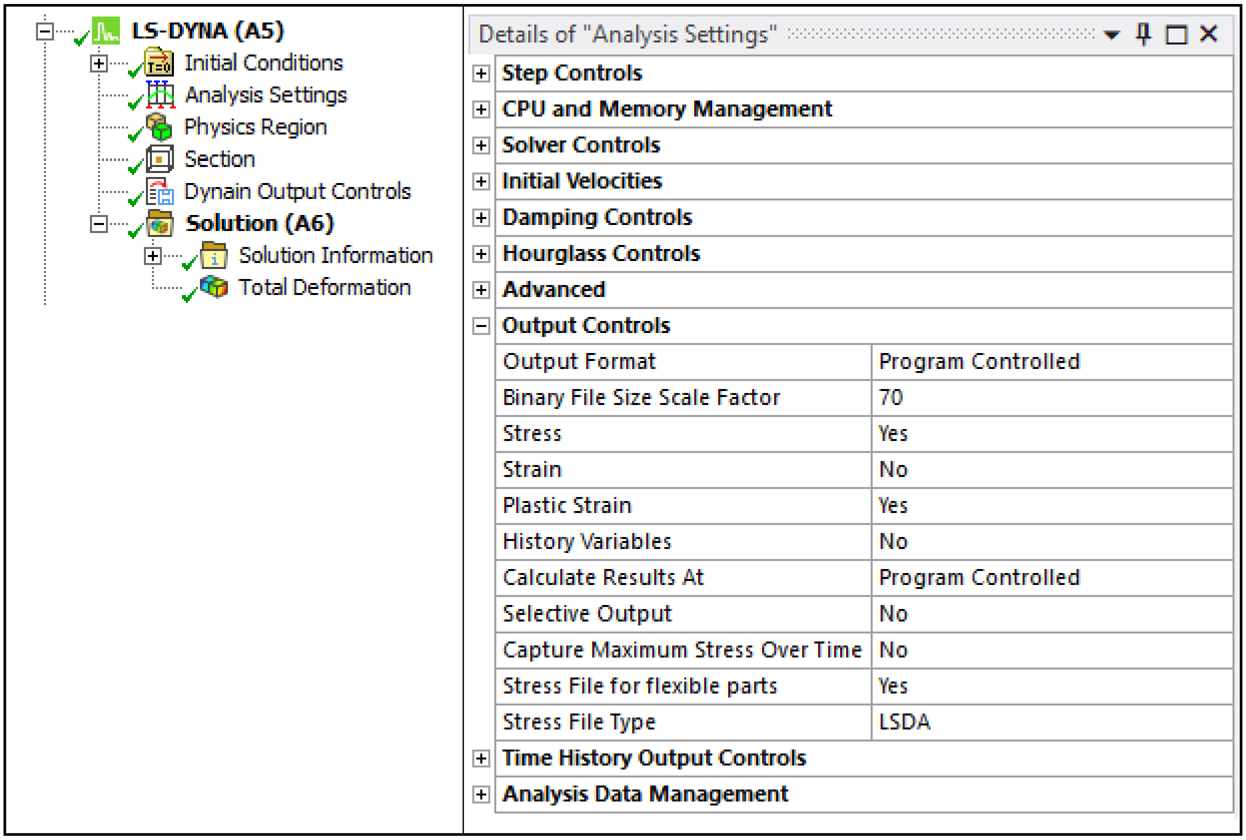
This section describes how to setup a two-system simulation using shared model data. In this case the Solution cell from the first system is linked to the Setup cell for the second system.



Linking two LS-DYNA systems in this way results in the automatic creation of a Pre-Stress initial condition in the Analysis for the second system. This indicates that data is to be transferred between the systems.



In order to enable the transfer of data a `dynain.lsda` file must be generated by the analysis of the first system. This is achieved by setting the property **Stress File for flexible parts** to Yes under the Output Controls section of Analysis Settings. By default, the dynain file is created in the LSDA format. There is an option to switch this to ASCII and Binary. However, the multi-system workflow does not currently support using either of those formats. The default settings are to transfer data for only flexible bodies. This behavior can be customized using the Dynain Output Controls (p. 108) object described below.

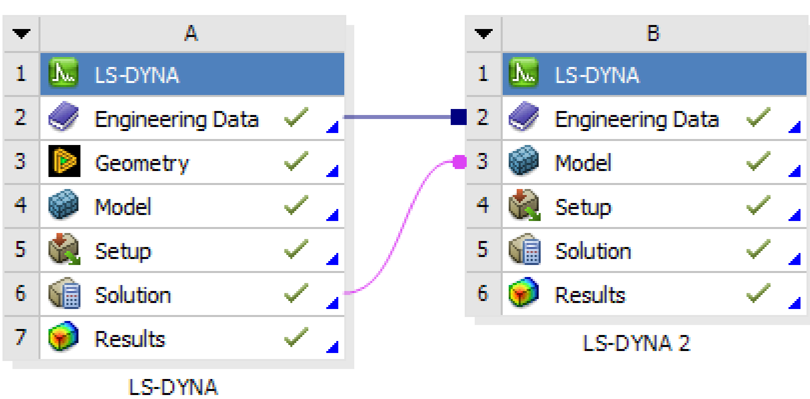


The `dynain.lsda` file is automatically copied from the system one analysis to the system two analysis and submitted to the solver via the *INCLUDE keyword in the system two input file.

This procedure can be extended to include additional systems as required.

3.7.14.2. Setting up a Multi-System Analysis using Solution Transfer

This section describes how to setup a two-system simulation using the results of the first solution to initialize the model of the second solution. In this case the Solution cell from the first system is linked to the Model cell for the second system.



In order to properly transfer the deformed geometry and stresses, you need to select the **Process Solution Data** check box in the Properties of the first LS-DYNA system. The data can only be transferred at the end of the solution, not at an earlier point in the solution.

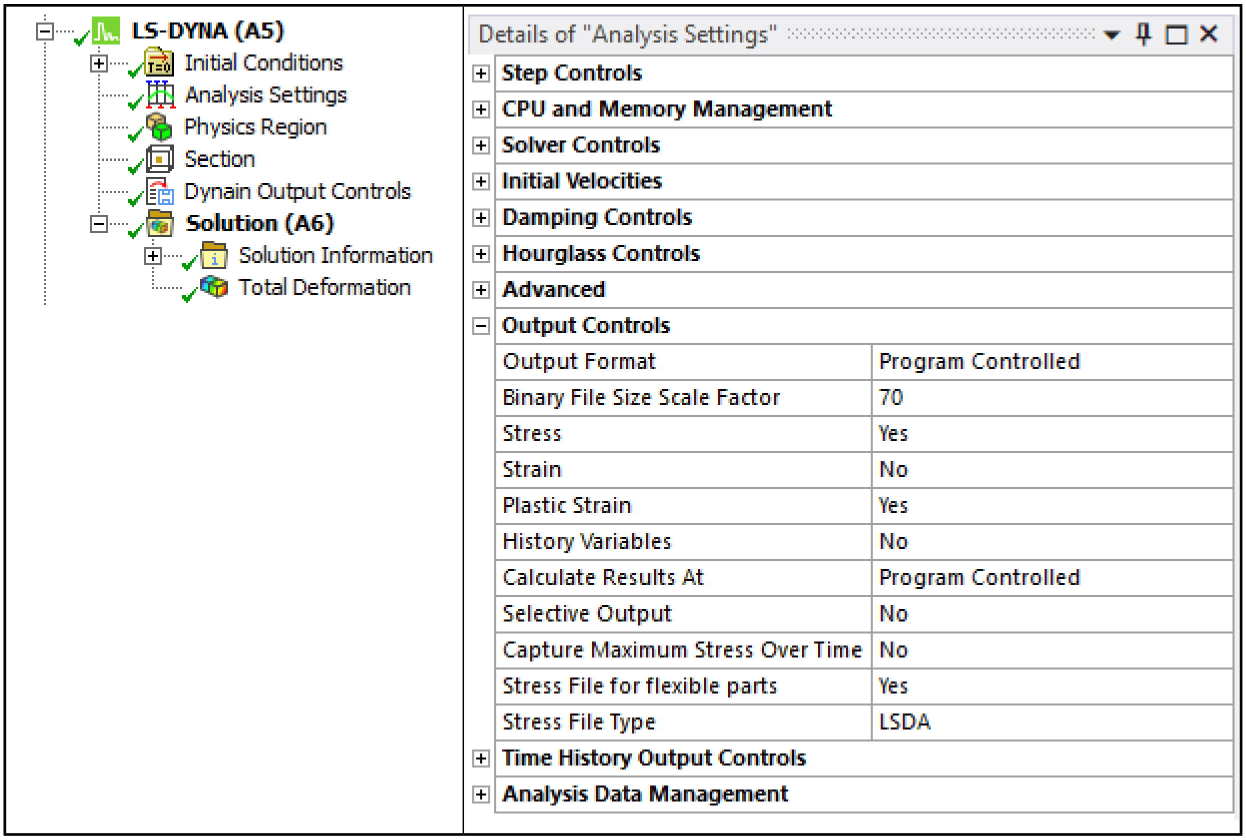
Properties of Schematic A6: Solution

	A	B	D
1	Property	Value	P
2	⊞ General		
6	⊞ Notes		
8	⊞ Used Licenses		
10	⊞ System Information		
14	⊞ Solution Process		
17	⊟ Update Settings for LS-DYNA (Component ID: Model 1)		
18	Process Nodal Components	☑	
19	Nodal Component Key		
20	Process Element Components	☑	
21	Element Component Key		
22	Process Solution Data	☑	

Note:

The unit systems used by the downstream systems must be the same as the unit system of the upstream system. If the unit systems are different an error is generated.

In order to enable the transfer of data a `dynain.lsda` file must be generated by the analysis of the first system. This is achieved by setting the property **Stress File for flexible parts** to Yes under the Output Controls section of Analysis Settings. By default, the dynain file is created in the LSDA format. There is an option to switch this to ASCII and Binary. However, the multi-system workflow does not currently support using either of those formats. The default settings are to transfer data for only flexible bodies. This behavior can be customized using the Dynain Output Controls (p. 108) object described below.

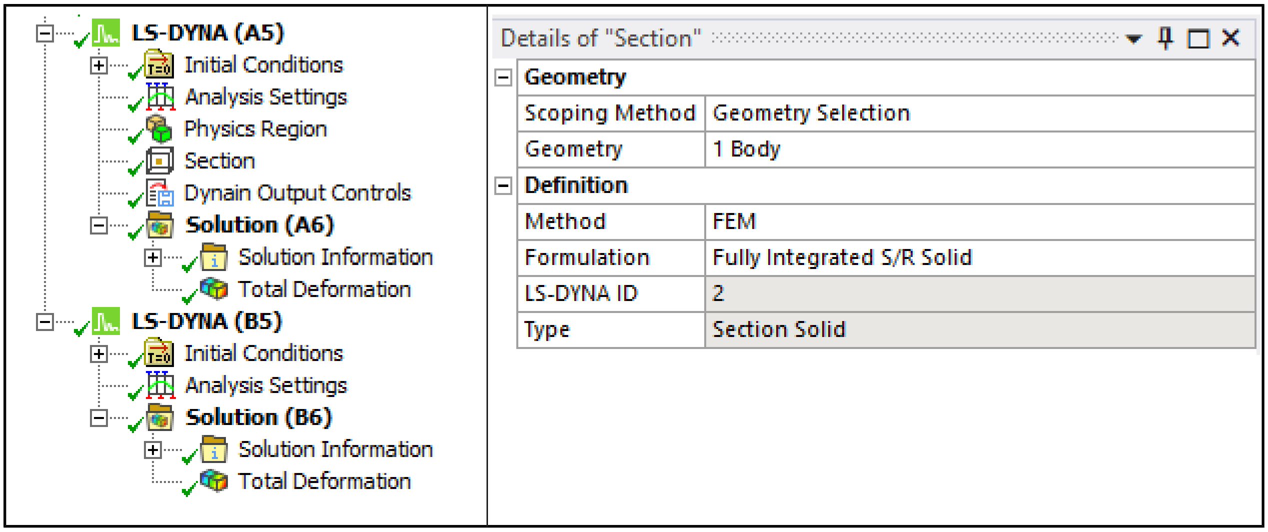


The `dynain.lsda` file is automatically copied from the first analysis system to the second analysis system and submitted to the solver via the *INCLUDE keyword in the system two input file.

This procedure can be extended to include additional systems as required.

3.7.14.3. Body Settings Automatically Transferred Between Systems

In order to allow for continuity between systems, any Section definitions are automatically transferred to subsequent analyses. For example, a custom element formulation defined in the first analysis is automatically written to the second analysis input file without it having to be explicitly defined in that analysis. If a Section definition is added to the second analysis it will override the Section definition from the first analysis, effectively changing the element formulation between analyses.

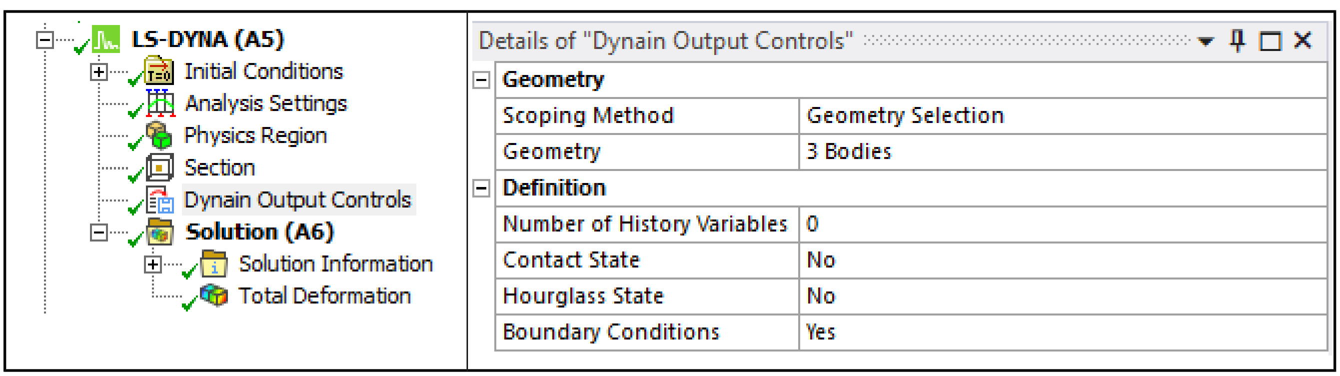


All other analysis-specific settings need to be explicitly defined within each analysis.

3.7.14.4. Customizing the Data Transferred Between Systems

The data that can be transferred via the `dynain.lsda` file is defined using the *INTERFACE_SPRING-BACK_LSDYNA and *INTERFACE_SPRINGBACK_EXCLUDE input file keywords.

Some of the properties for the input cards can be customized using the Dynain Output Controls object. This includes the bodies that will be transferred, allowing the default of only flexible bodies to be overridden to all bodies or any subset. Beams are supported in this workflow. Other settings that can be controlled include the history variables that are transferred, contact and hourglass state at the end of the previous analysis, and time-independent boundary conditions (*BOUND-ARY_SPC_NODE).



The other settings for these input file keywords take their default values. Refer to the LS-DYNA Keyword Manuals for details.

3.7.15. Splitting the LS-DYNA Input File

The Split Input File option allows you to save the different sections of the input file in separate files. An input.k file will be generated after solving and it will import the additional .k files generated by the Split Input File option. This is accomplished using the *INCLUDE command as seen in the following figure.

```
$# LS-DYNA Keyword file created by WB-LSDYNA 2024 R2
*KEYWORD
*PARAMETER_DUPLICATION
4
*INCLUDE
database.k
*INCLUDE
control.k
*INCLUDE
mesh.k
*INCLUDE
contact.k
*INCLUDE
component.k
```

The input files generated are:

`database.k`: keywords written to obtain output files containing results information

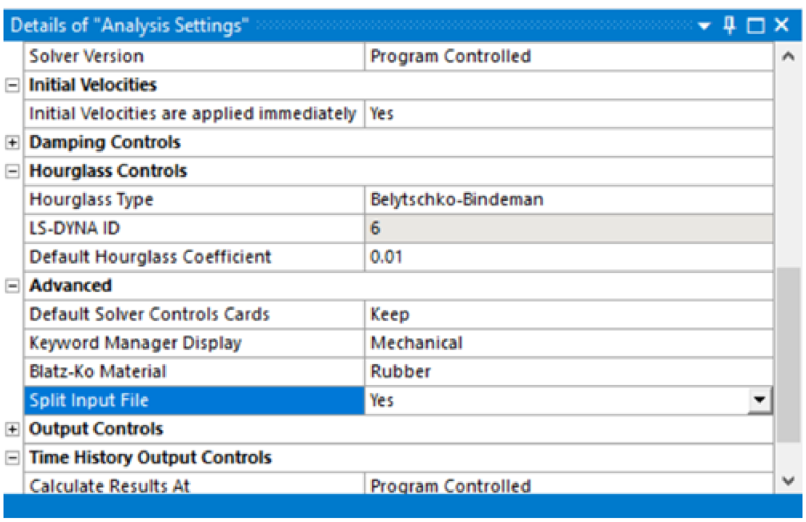
`control.k`: keywords written to define solution parameters and options

`mesh.k`: keywords written for the mesh

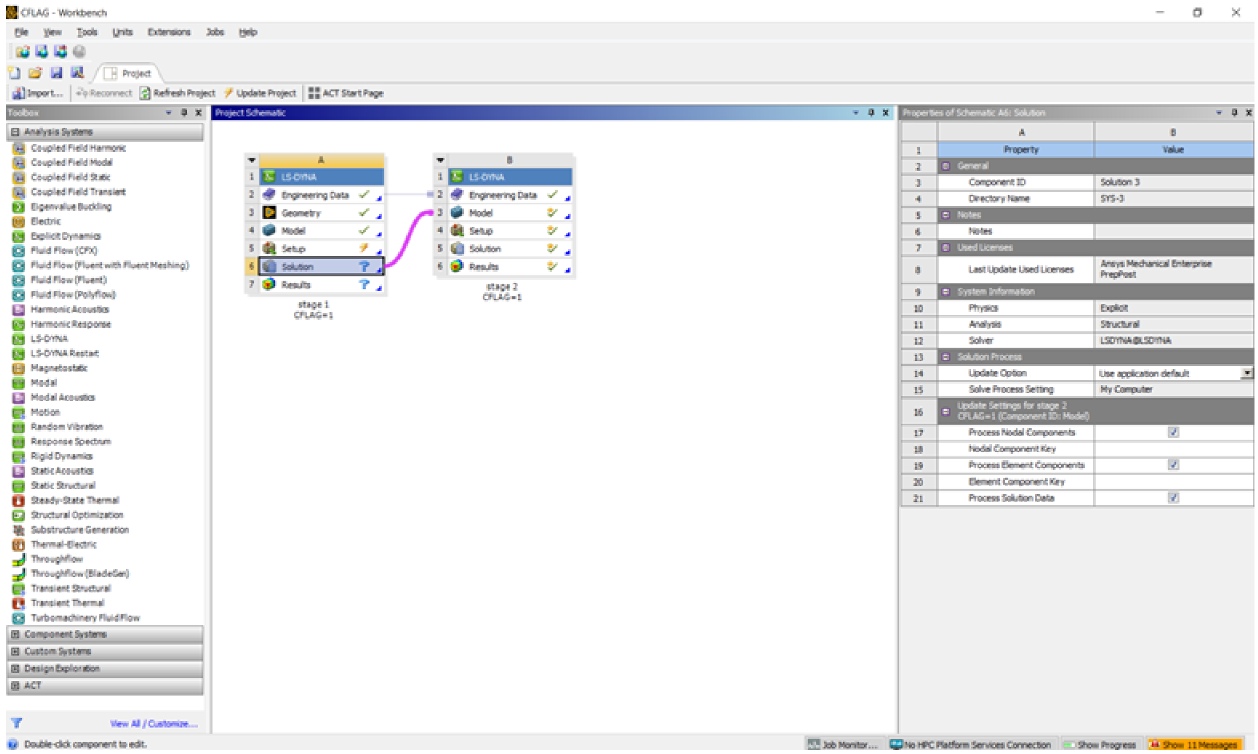
`component.k`: keywords written to define mesh components

`contact.k`: keywords written to define contacts between parts

Enable this capability by setting the **Split Input File** option to Yes in the **Advanced** category of the Analysis Settings Details panel.



You must split the input file if you want to preserve the contact IDs between systems when using an LS-DYNA system in a downstream Workbench analysis. In order to get the transfer to work properly you must right click the Solution cell of the upstream system, select Properties and in the Properties pane select the **Process Solution Data** check box.



3.8. Restarting an LS-DYNA Analysis

Restarting means performing an analysis which continues from a previous analysis. A restart can begin from either the conclusion of or the middle of a prior analysis.

Possible Reasons for Performing A Restart

- The previous analysis was killed by the operating system or the user (sw1).
- The previous analysis exceeded the user defined CPU limit.
- There was an error in the previous analysis and a restart is used to diagnose and/or correct the error.
- The previous analysis was not run to a long enough termination time.

There are three types of restarts: simple restarts, small restarts, and full restarts.

A *simple restart* is one for which the original model has not been altered in the new analysis. A simple restart is performed when the LS-DYNA solution was prematurely interrupted by the exceeding of a user defined CPU limit or by the issuing of the sense switch control sw1.

A *small restart* is used to run an analysis to a longer termination time than initially specified and/or to make minor modifications to the model. The following actions are permitted in a small restart.

- Specifying rigid/deformable switch controls.
- Switching parts from deformable to rigid & back.

A *full restart* supports most new analysis actions, including:

- Portions of the model may be added or removed.
- Additional materials and loading changes are permitted.

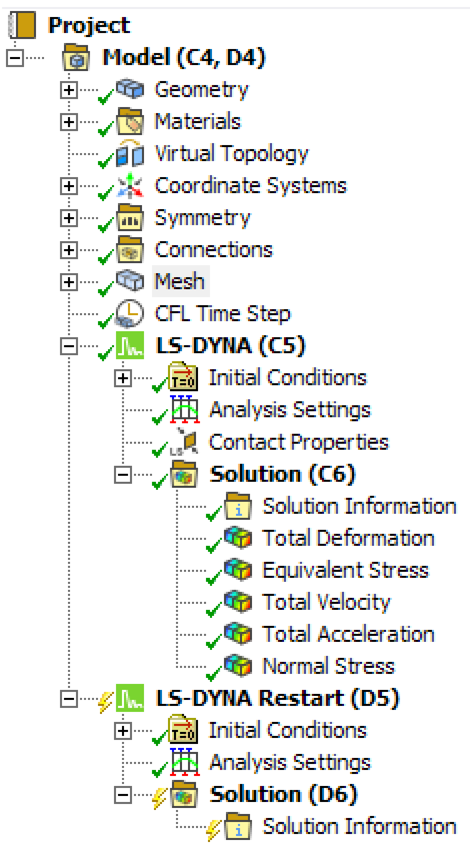
There are some restrictions for full restarts, including:

- Contact specifications and initial velocities cannot be changed.
- Adaptive meshing is not supported, even if present in the initial run.

Stress initialization is available for full restarts. Deformed nodal positions and stresses/strains from a previous analysis are carried forward into a full restart analysis.

Note:

In order to switch the stiffness in either a small restart or full restart, you must add a **Deformable To Rigid** object under the original LS-DYNA system as shown here:

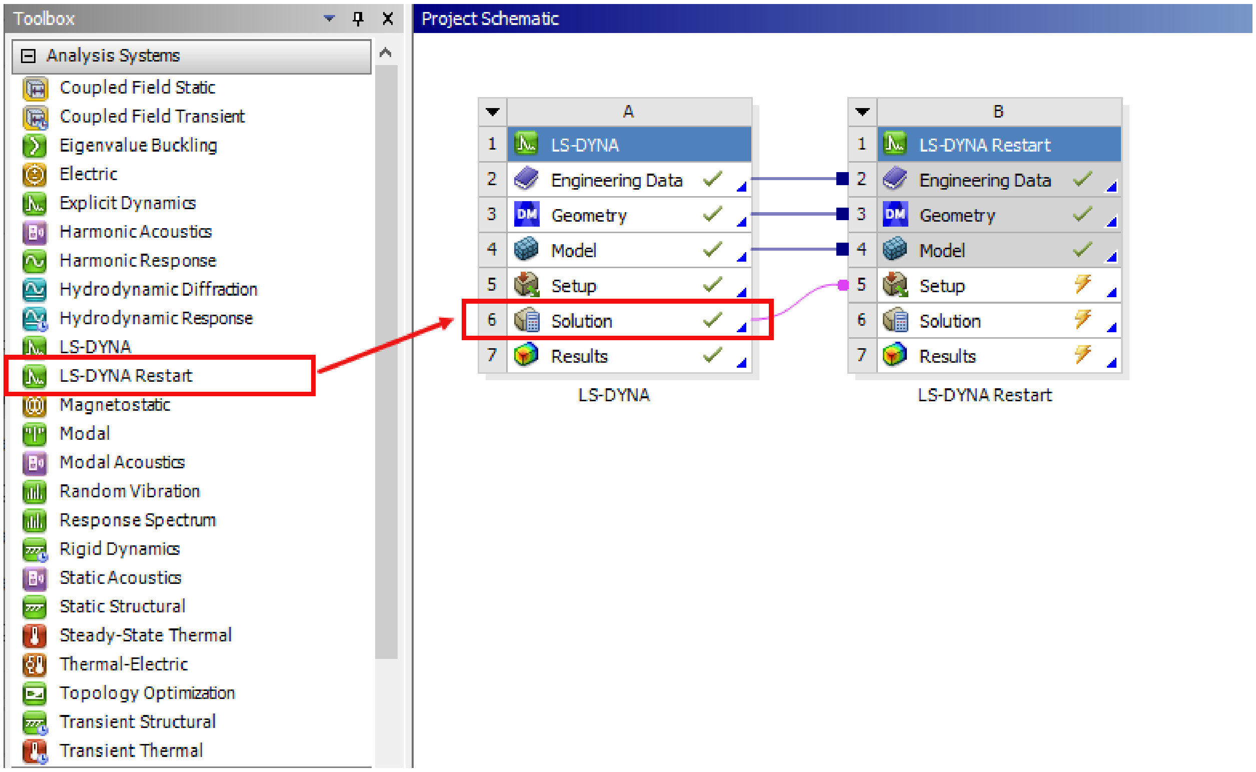


To add the object, select the **LS-DYNA** object, right-click and select **InsertDeformable To Rigid**. After you add the object, select it and in the **Details** view, scope the object to the body whose stiffness you want to change. Be sure that the **Stiffness Behavior** attribute of the scoped body is set to Flexible.

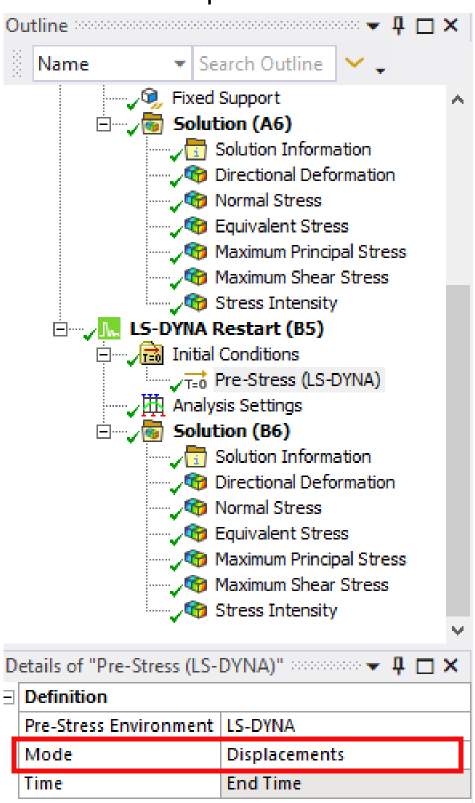
3.8.1. Performing a Simple Restart

In order to perform a simple restart:

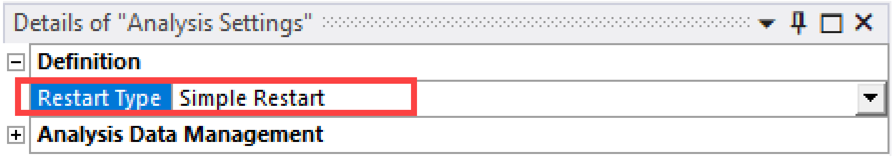
1. On the Project Schematic page, select LS-DYNA Restart from the Toolbox and drag and drop it onto the **Solution** cell of an existing LS-DYNA system.



2. Under the **LS-DYNA Restart** object, select **Pre-Stress (LS-DYNA)** under **Initial Conditions** and set **Mode** to Displacements in the Details panel.



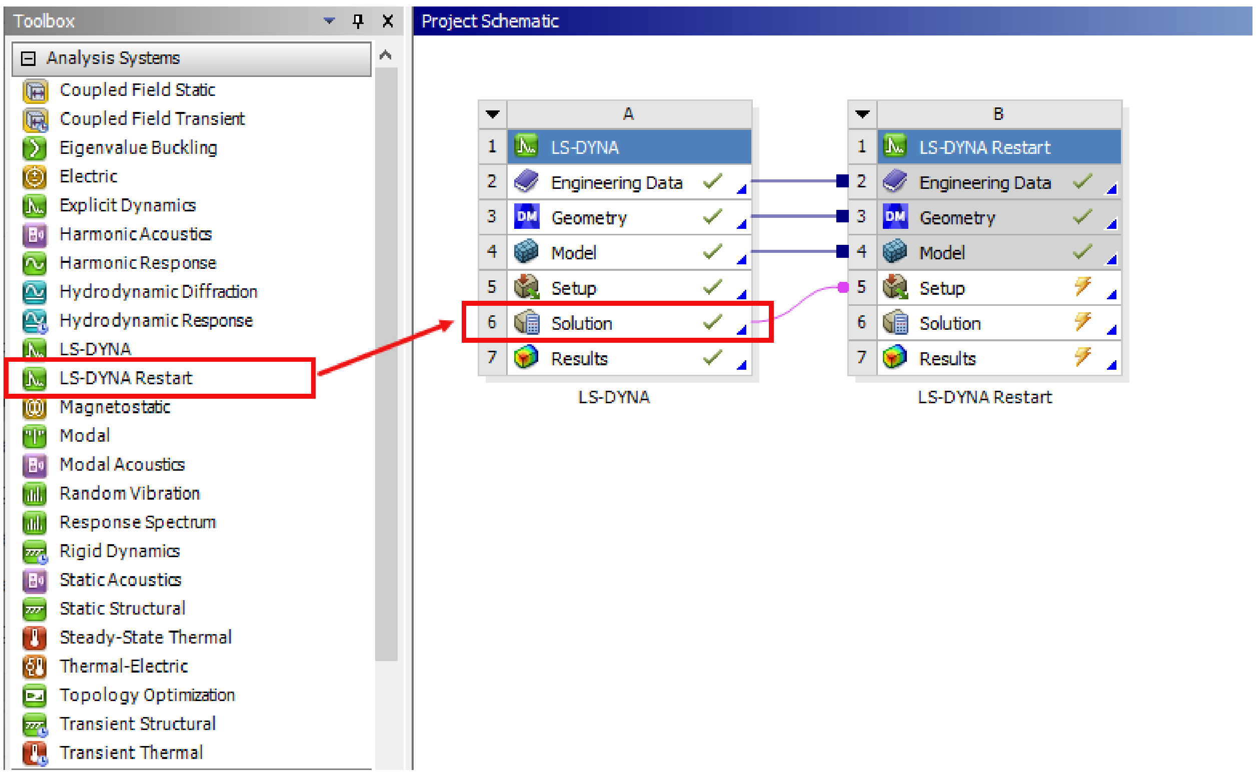
3. In the Details panel of the **Analysis Settings** object, choose Simple Restart for **Restart Type** and then Solve.



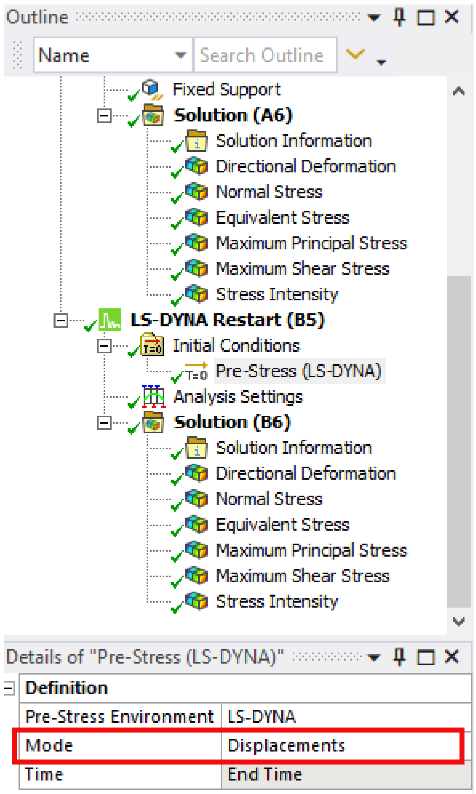
3.8.2. Performing a Small Restart

In order to perform a Small Restart:

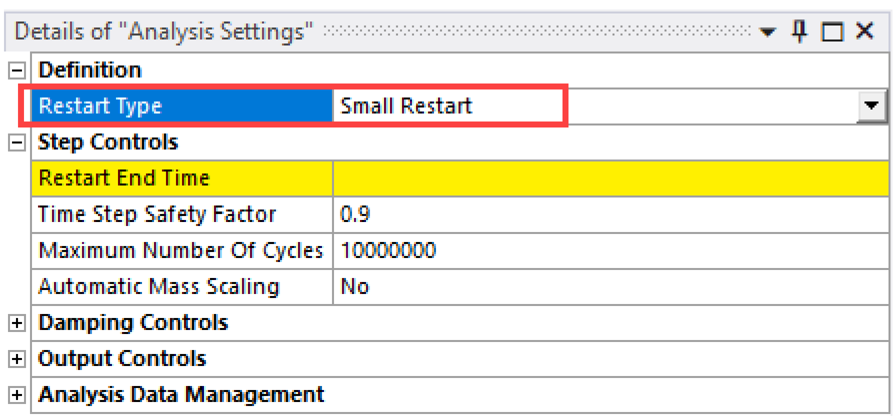
1. On the Project Schematic page, select LS-DYNA Restart from the Toolbox and drag and drop it onto the **Solution** cell of an existing LS-DYNA system.



2. Under the **LS-DYNA Restart** object, select **Pre-Stress (LS-DYNA)** under **Initial Conditions** and set **Mode** to Displacements in the Details panel.

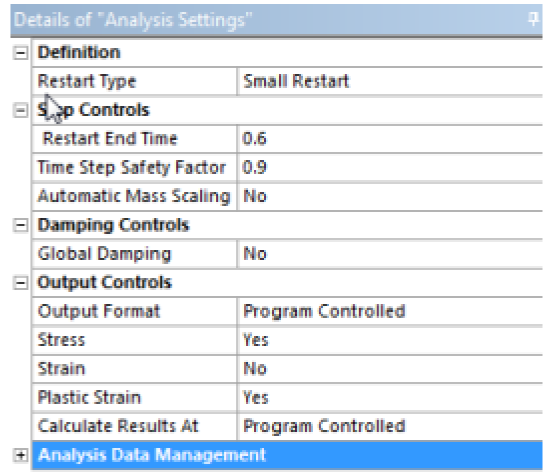


3. In the Details panel of the **Analysis Settings** object, choose Small Restart for **Restart Type** and define a new termination time.



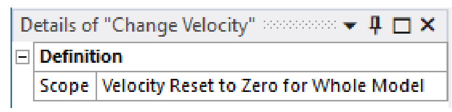
4. Make any additional changes to the project. The allowed changes are:

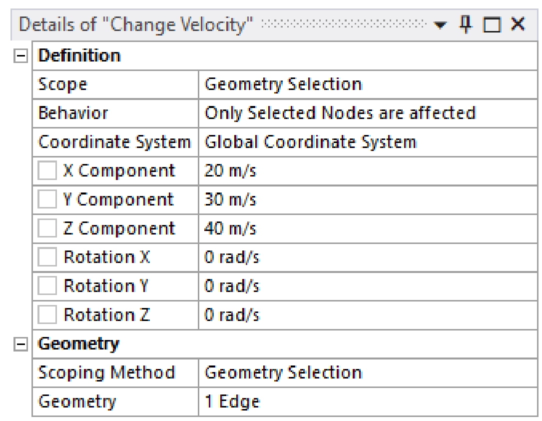
Actions	Tab Selection	Details Panel
• Reset termination time. • Reset output printing interval. • Reset output plotting interval. • Change damping options.	None	See Details of "Analysis Settings" below

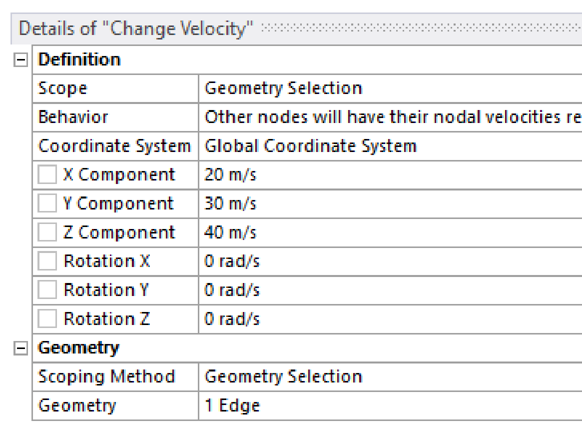


• Change velocity options. (p. 222)

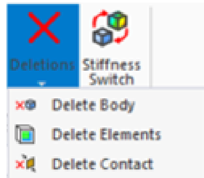
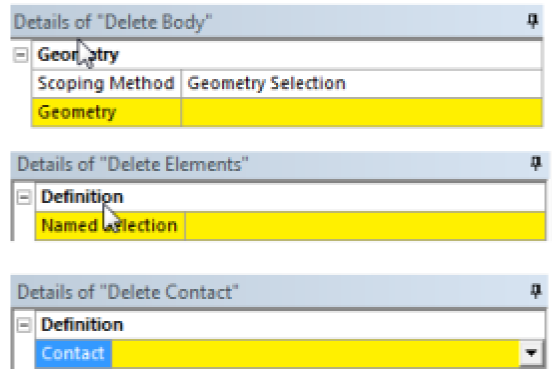
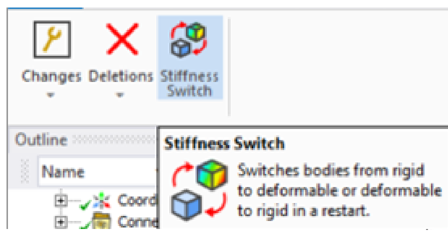
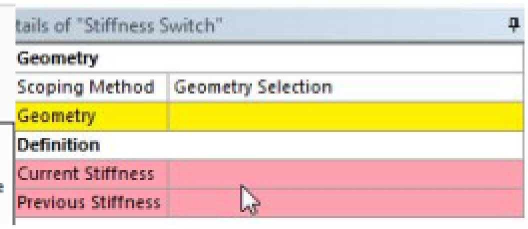
Three Change Velocity Options:







A coordinate system can be selected for the Change Velocity object, and it can be scoped to

Actions	Tab Selection	Details Panel
		vertices, edges, faces, or bodies.
• Delete contact surfaces. (p. 224) • Delete elements and parts. (p. 224)		**Three possibilities:** 
• Switch deformable bodies to rigid. • Switch rigid bodies to deformable.		

For more information about actions you can take during a Small Restart see Restart (p. 222).

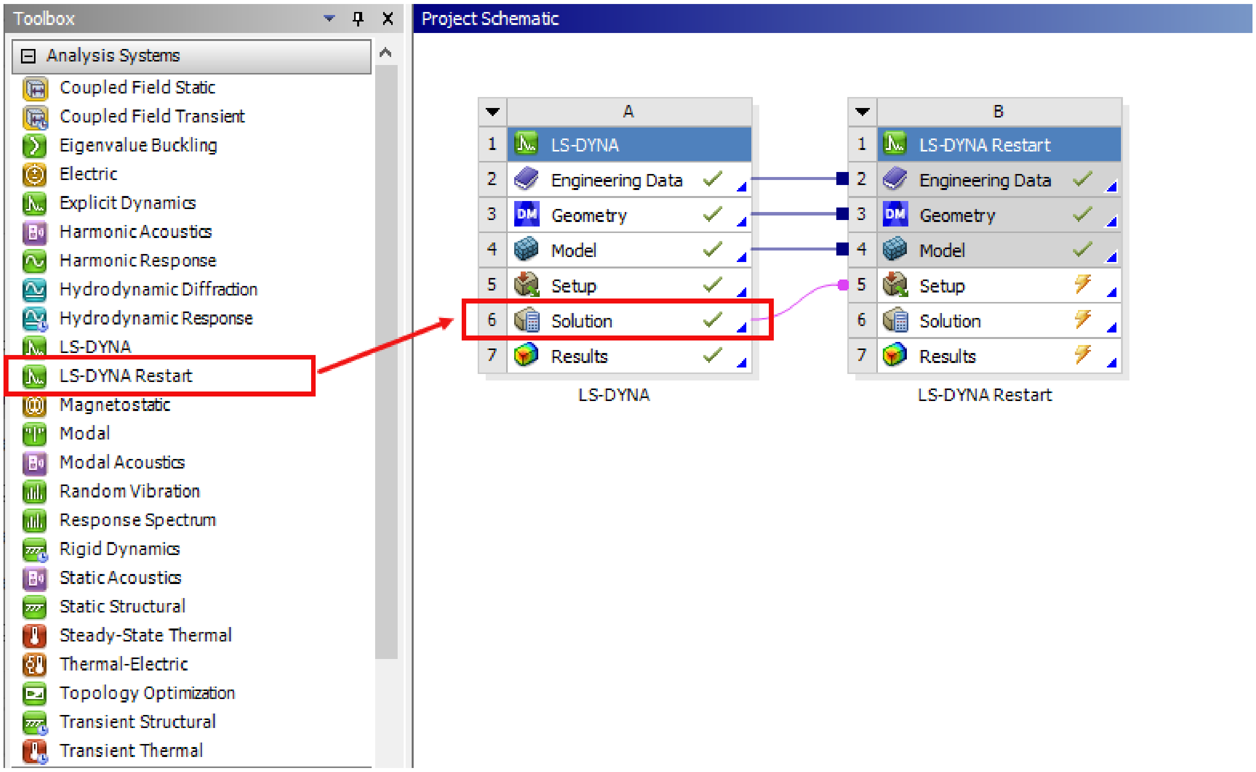
Note:

A small restart cannot be done if you are running a serial solution in single precision. To do a small restart you must be running a parallel solution or using double precision.

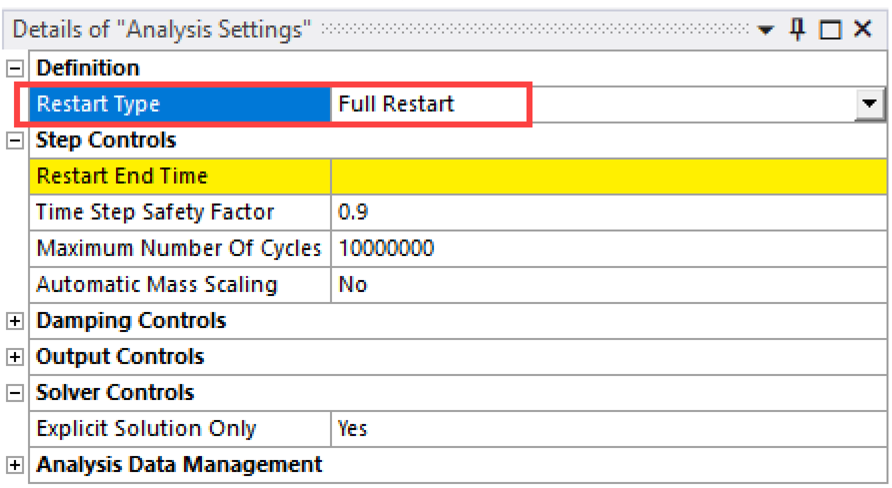
3.8.3. Performing a Full Restart

A full restart is a new analysis starting from an initialized state. New data may be entered into the model, including nodes, elements, material data, and loading.

1. On the Project Schematic page, select LS-DYNA Restart from the Toolbox and drag and drop it onto the **Solution** cell of an existing LS-DYNA system.



2. In the Details panel of the **Analysis Settings** object, choose Full Restart for **Restart Type**.

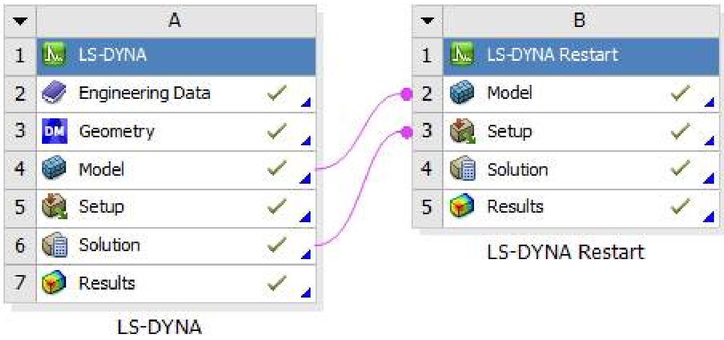


3. Make any other changes to the project that you require and then Solve.

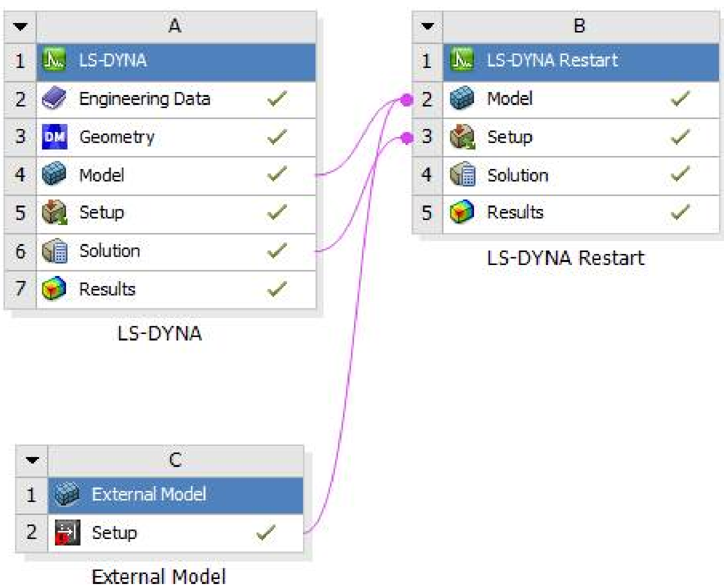
Note:

A full restart begins at the point in time where the previous calculation ended. Any new time-dependent loading applied during a full restart must start after the physical time that elapsed in the previous calculation. For example, if you want to ramp a velocity from 10 m/s to 20 m/s in the full restart for a total duration of 2ms and the previous calculation ended at 1ms, the loading should have a point in time at 1ms with a value of 10 m/s, and a point in time at 3ms with a value of 20m/s.

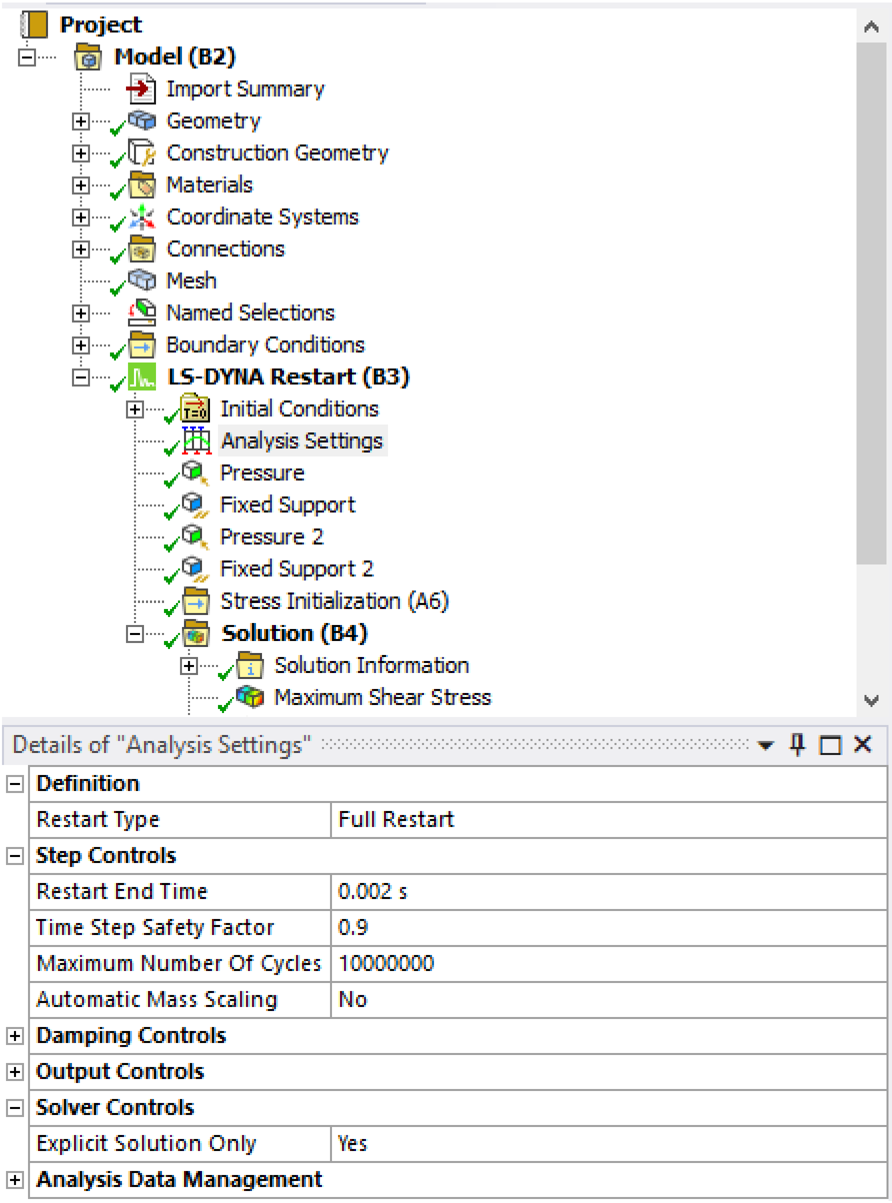
You can add new data into the model when doing a full restart, including nodes, elements, and material data. The full restart can be combined with the model assembly features of Mechanical to add new bodies to an already solved model, or to modify it geometrically while keeping the initialization of the first simulation.



In the workflow above, the mesh and material from system A is transferred as a new model to system B. The Restart system (B) is a separate analysis, allowing you to make geometric changes (transformations) to the initial geometry.



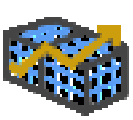
In this second workflow, the initial model is transferred to the full restart system, and combined with new mesh data from External Model. This new mesh could also come from a Mechanical model.



You can also do a restart from an implicit solution with the LS-DYNA solver (p. 11). In this case, set **Explicit Solution Only** to No.

3.9. Additional LS-DYNA Analysis Tools

When the LS-DYNA system is selected in the Outline view, you will see these additional analysis tools on the LSDYNA Pre tab:

-  - use this tool to define the time step for the analysis of rigid body dynamics.

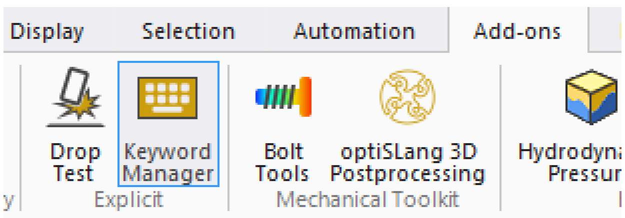
-  - use this tool to calculate CFL time step before the calculation starts.

Chapter 4: The LS-DYNA Keyword Manager

The LS-DYNA Keyword Manager allows you to enter LS-DYNA keywords as objects in the project tree when an LS-DYNA system is present. This provides an easy way to insert keywords that cannot be generated by the Mechanical system into the `.k` file generated by the system. The options for the individual keywords are entered using the fields in the **Details** panel for each keyword object.

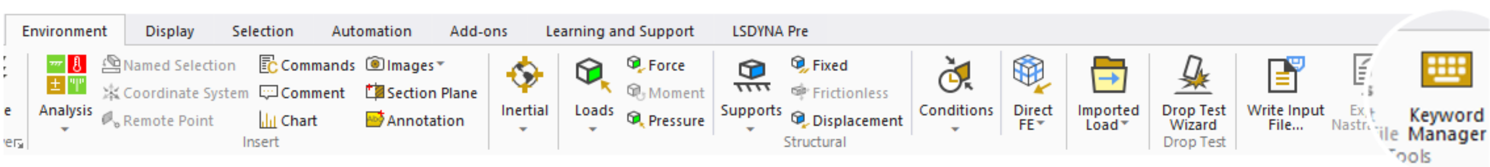
4.1. Activating the Keyword Manager

The Keyword Manager is an add-on that provides for advanced usage of LS-DYNA keywords from within the Mechanical user interface. It is not activated by default. To load the add-on, simply click the icon in the Add-ons Ribbon.

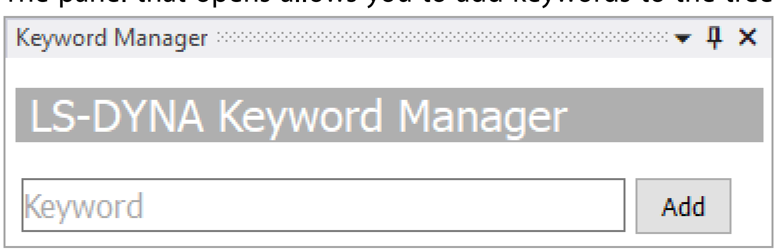


4.2. Using the Keyword Manager

The Keyword Manager is opened by clicking the Keyword Manager icon in the Environment tab.

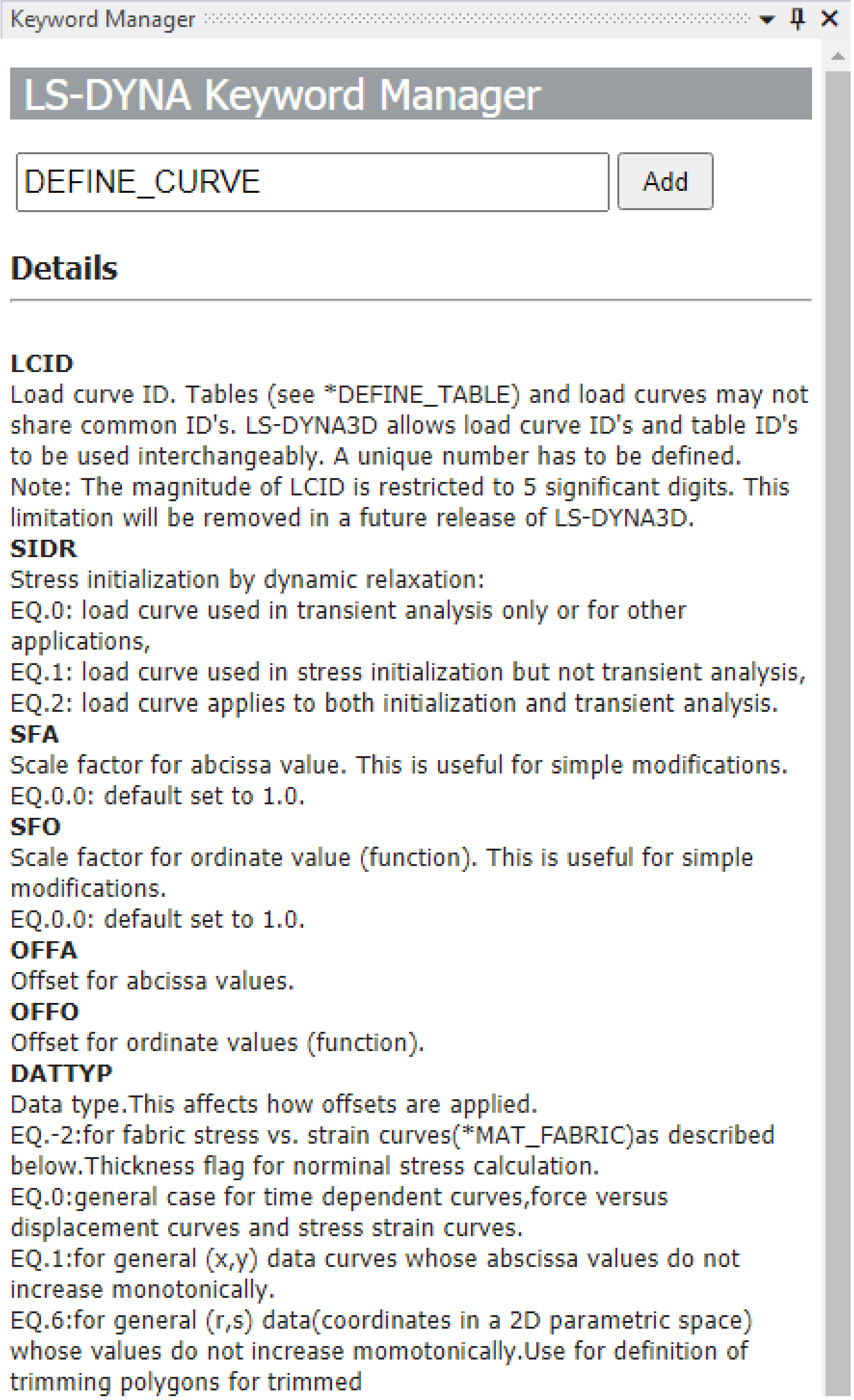


The panel that opens allows you to add keywords to the tree of the active Mechanical analysis.

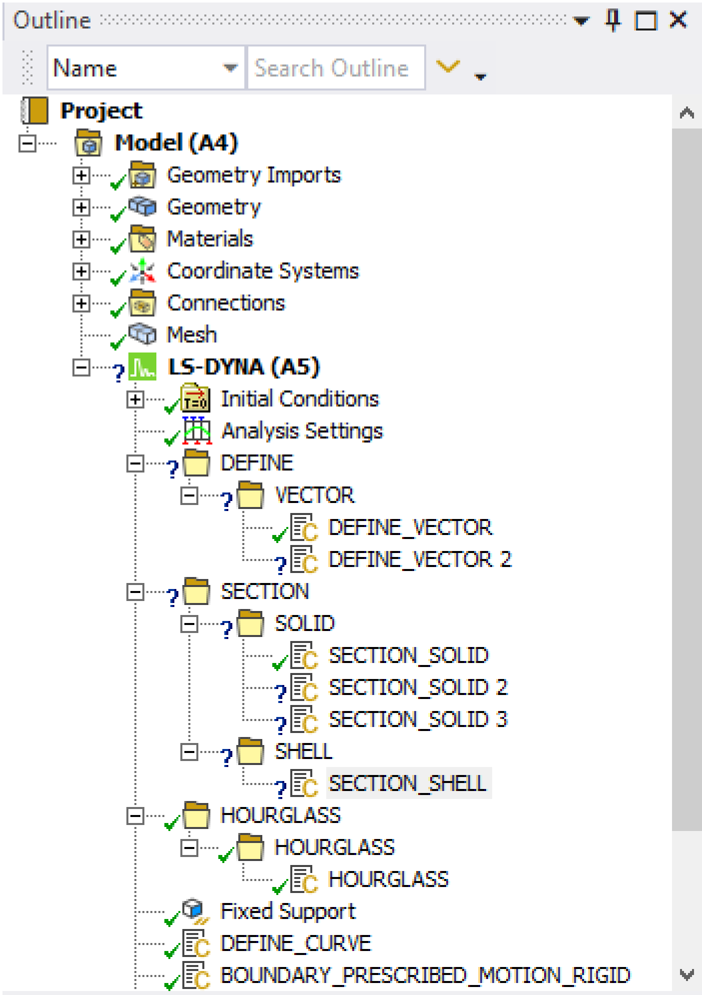


Most LS-DYNA keywords can be created, viewed, modified, and deleted as part of the model building process using the Keyword Manager.

To add a keyword, start typing the name of a keyword in the **Keyword Manager** panel to get a matching list of available keywords. Select the desired keyword. A description of each of the keyword's parameters is provided. This description is the same as in the LS-DYNA Keyword Manuals. To add the keyword object to the tree, click **Add**.

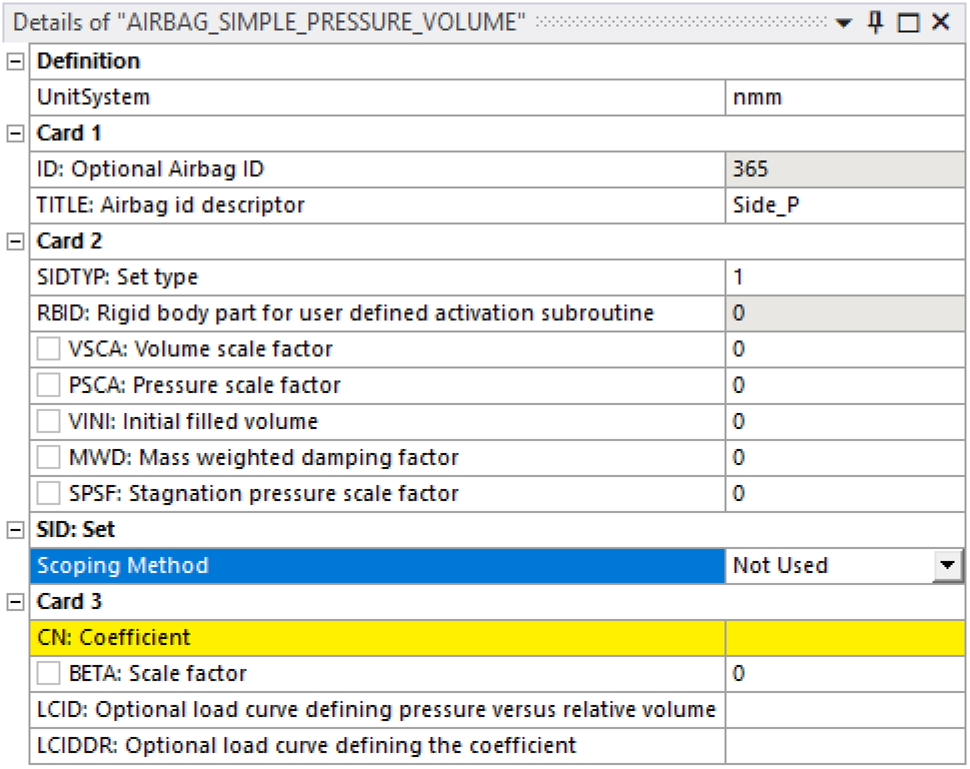


As you add keyword objects to the active analysis using the Keyword Manager, the keyword objects will be grouped by family name with subgroups for the individual keywords.



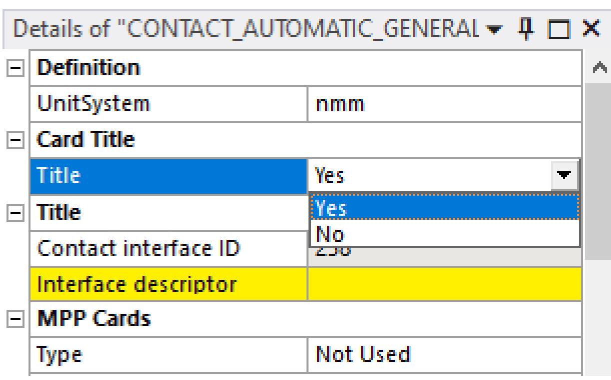
Each keyword object added to the tree is integrated into the input file generation process, and the corresponding input file keyword is written to the input file used in the calculation.

The **Details** view is specific to each keyword. The fields in the Details panel are grouped by the corresponding Cards for that particular Keyword. You have the ability to turn off options by selecting Not Used.

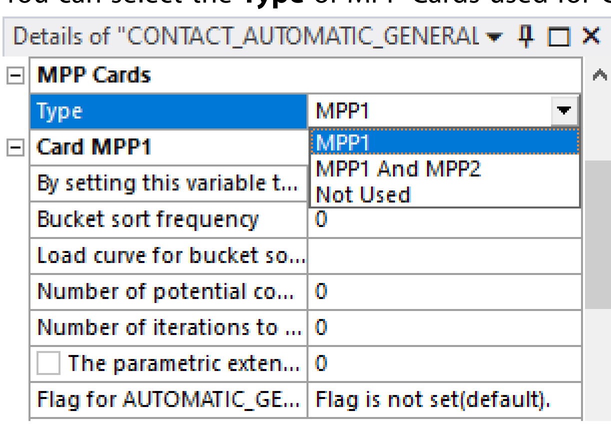


Note that the Contact keywords behave differently than other keywords.

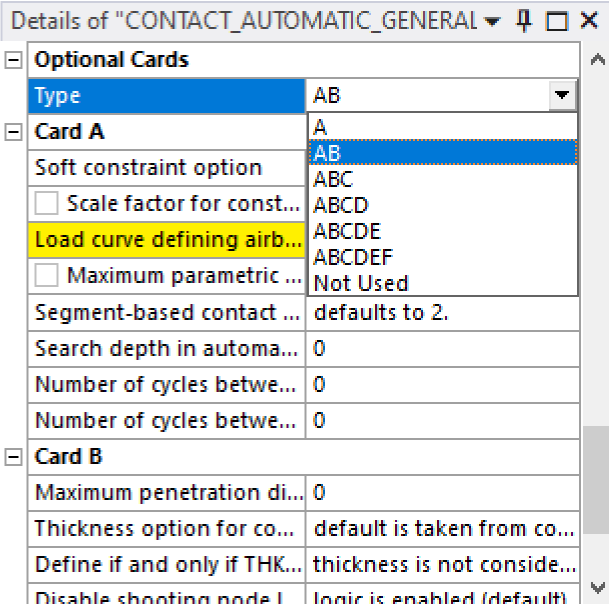
- The **Title** field, which is written to the LS-DYNA input file, can be turned off for Contact keywords.



- You can select the **Type** of MPP Cards used for Contact keywords (or omit them).

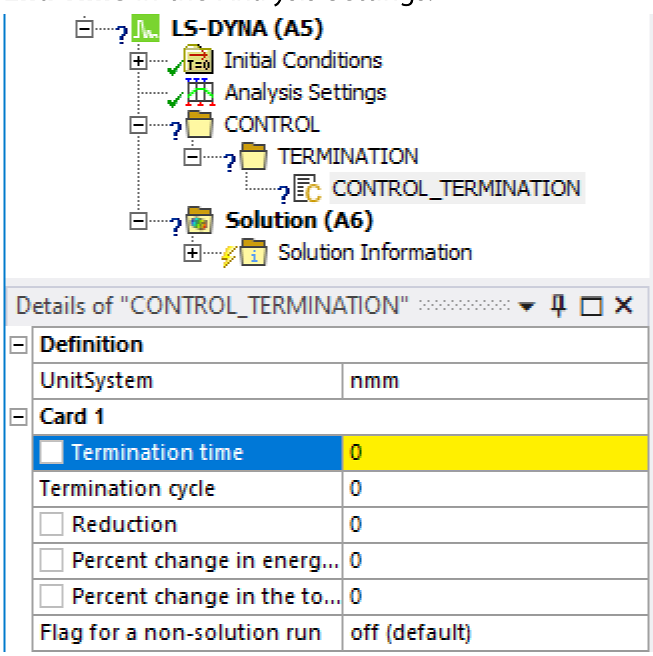


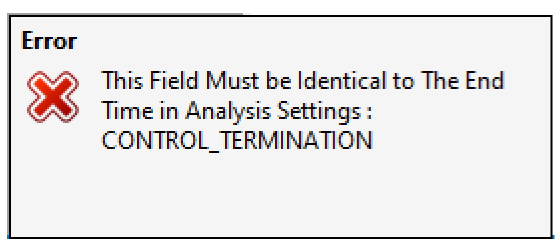
- You can specify whether additional optional Contact cards (A, B, C, D, E, F) are used.



Control Keyword Behavior

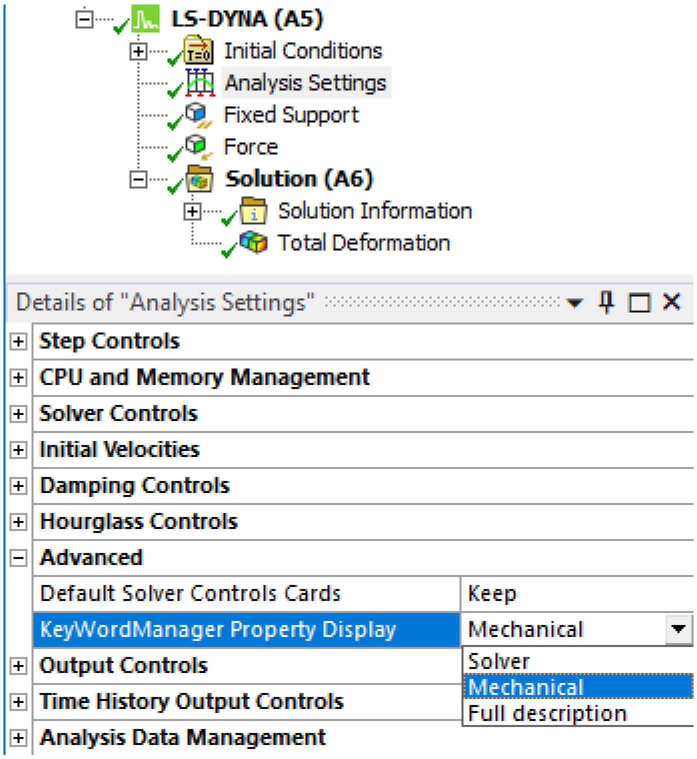
When you insert Control keywords with the Keyword Manager, they will override any Control keywords that are generated by the Analysis Settings. The only exception to this is that if you insert a CONTROL_TERMINATION keyword using the Keyword Manager, you must set **Termination time** to the same value as **End Time** in the Analysis Settings.





Keyword Property Display Names

You can change the manner in which the property names are displayed by setting the **KeyWordManager Property Display** field in the Analysis Settings. Choosing Mechanical uses the property names that are in the style of Mechanical property names. Choosing Solver uses the LS-DYNA variable names for the properties. Choosing Full description uses both styles at once for the property names.



4.3. Notes and Limitations

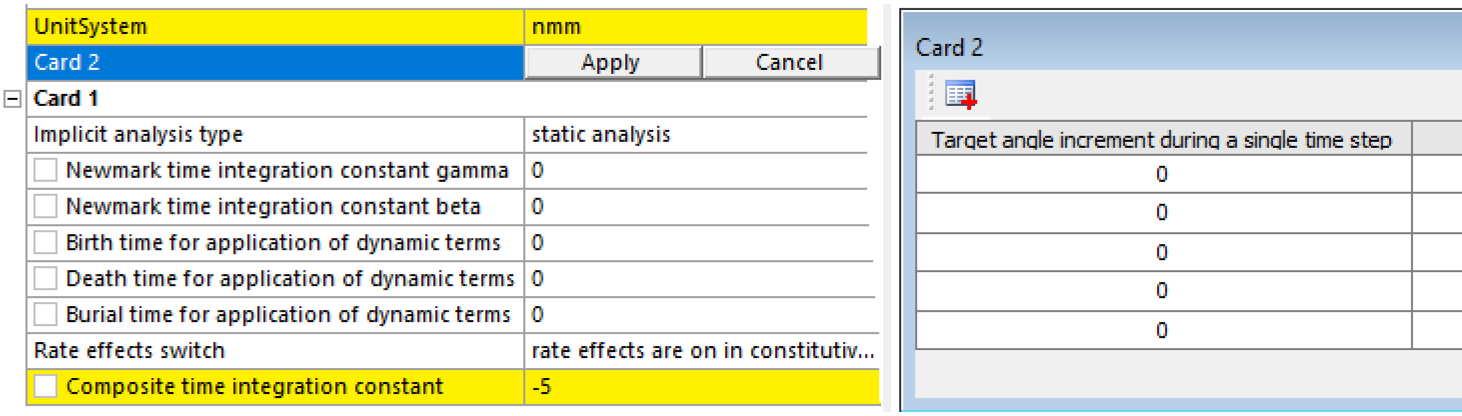
Notes on the Insertion of Keywords

- Keywords inserted after you change the setting will display the field names in the chosen manner. To change the way field names are displayed for keywords that were inserted prior to changing the display setting, you must exit Mechanical and reopen it.

- The Unit System of each keyword must be the same as the unit system of the corresponding calculation.

- In some objects, there is a Program Controlled option on fields that allows the Mechanical value of the fields to be used, while also providing other options. An example of this is the **Material defined in *Mat section** field in a Part. Setting this to Program Controlled uses the material that was defined for the Mechanical part that it is scoped to. Alternatively, you have the option to choose other materials that are defined in the system.

- For the CONTROL_IMPLICIT_DYNAMICS keyword, if the **Composite time integration constant** is 0, only Card 1 is used. To include Card 2, enter a negative number equal to the number of rows you want for Card 2 into the field. You must manually add the rows to the table that appears when you click on the Card 2 entry field.



- In order to use any of these SPH keywords:
  - DEFINE_SPH_INJECTION
  - DEFINE_ADAPTIVE_SOLID_TO_DES
  - DEFINE_SPH_MESH_BOX
  - DEFINE_SPH_MESH_SURFACE

  You must add a Part keyword and set the **Scoping Method** of the Part to Program Controlled. Then, for each of the keywords listed above that you define, select a defined Part in the **Part ID for generated SPH elements** field. This allows the solver to create a virtual part to be used in the SPH analysis.

- Some keywords can be inserted but may be not fully implemented or tested. These keywords have **(beta)** appended to their name in the Keyword Manager. You can still insert these keywords, but they are not officially supported. See for discussion of issues for specific keywords that are beta.

Chapter 5: Keywords used by LS-DYNA in Workbench

The LS-DYNA Workbench system allows you to run an explicit dynamics analysis for your model using the LS-DYNA solver.

This section shows all LS-DYNA supported keywords and their syntax. Keywords conform to the LS-DYNA Keyword User's Manual Volumes I, II, III .

Each keyword consists of one or more cards, each with one of more parameters. If a parameter is not shown, it will be assigned default values by the LS-DYNA solver.

This chapter describes the following:

5.1. Input File Header

*KEYWORD

Marks the beginning of a keyword file.

5.2. Database Format

*DATABASE_FORMAT

Specifies the format in which to write binary results files like D3PLOT and D3THDT.

Card

- IFORM = 0. Binary results will be written only in the LS-DYNA format.
- IFORM = 2. Both LS-DYNA and Ansys database formats will be written.
- IBINARY = 0. Word size of the binary output files (D3PLOT, D3THDT, ...) defaults to 64 bit format.

5.3. Control Cards

*CONTROL_ACCURACY

Specifies control parameters that can improve the accuracy of the calculation.

Solver Controls	
Solver Type	Program Controlled
Solver Precision	Program Controlled
Unit System	nmm
Invariant Node Numbering	Off
Second Order Stress Update	No

Card

- OSU is the global flag for objective stress updates. Required for parts that undergo large rotations. This value is set to 1 when the value of the **Second Order Stress Update** from the Solver Controls section of the **Analysis Settings** is set to On. Otherwise it is set to 0.
- INN is the invariant node numbering for shell and solid elements. This value is set based on the **Invariant Node Numbering** from the Solver Controls section of the **Analysis Settings** .
  - = -4 if the Invariant Node Numbering is set to On for both shell and solid elements except triangular shells.
  - = -2 if the Invariant Node Numbering is set to On for shell elements except triangular shells.

– = 1 if the Invariant Node Numbering is set to Off.

– = 2 if the Invariant Node Numbering is set to On for shell and thick shell elements.

– = 3 if the Invariant Node Numbering is set to On for solid elements.

– = 4 if the Invariant Node Numbering is set to On for shell, thick shell, and solid elements.

*CONTROL_ALE

Inputs to this keyword are available in **Analysis Settings** object. Set global control parameters for the Arbitrary Lagrange-Eulerian (ALE) and Eulerian calculations.

ALE Controls	
Continuum Treatment	Use Alternate Advection Logic
Cycles Between Advection	1
Advection Method	Donor Cell + Half Index Shift
Simple Average Weighting Factor	-1
Volume Weighting Factor	0
Isoparametric Weighting Factor	0
Equipotential Weighting Factor	0
Equilibrium Weighting Factor	0
Advection Factor	0
Start	0 s
End	1E+20 s
Reference Pressure	0 Pa
Pressure Equilibrium Iteration	No

Card 1

- DCT = Flag to invoke alternate advection logic.
  - = -1 if the Continuum Treatment is set to Use Alternate Advection Logic (default).
  - <> -1 if the Continuum Treatment is set to Use Default Advection Logic.
- NDV = **Cycles Between Advection** from the ALE Controls section of the **Analysis Settings** (Default to 1).
- METH = **Advection Method** from the ALE Controls section of the Analysis Settings (Default to 1):
  - = 1 if the **Advection Method** is set to Donor cell + Half-Index-Shift.
  - = 2 if the **Advection Method** is set to Van Leer + Half-Index-Shift.
  - = -2 if the **Advection Method** is set to Van Leer (Relaxed Monotonicity).
  - = 3 if the **Advection Method** is set to Donor Cell (Total Energy Conservation).
  - = 6 if the **Advection Method** is set to Finite Volume With Flux Corrected Transport.
- AFAC = **Simple Average Weighting Factor** from the ALE Controls section of the **Analysis Settings**. Smoothing weight factor - Simple average.

- BFAC = **Volume Weighting Factor** from the ALE Controls section of the **Analysis Settings**. Smoothing weight factor - Volume Weighting
- CFAC = **Isoparametric Weighting Factor** from the ALE Controls section of the **Analysis Settings**.
- DFAC = **Equipotential Weighting Factor** from the ALE Controls section of the **Analysis Settings**.
- EFAC = **Equilibrium Weighting Factor** from the ALE Controls section of the **Analysis Settings**.

Card 2

- AAFAC = **Advection Factor** from the ALE Controls section of the **Analysis Settings** .
- START = **Start** from the ALE Controls section of the **Analysis Settings**.
- END = **End** from the ALE Controls section of the **Analysis Settings**.
- PRIT = **Pressure Equilibrium Iteration** from the ALE Controls section of the **Analysis Settings**.
- REF = **Reference Pressure** that is applied to the free surfaces of the ALE domain, from the ALE Controls section of the **Analysis Settings**.

*CONTROL_BULK_VISCOSITY

Sets the bulk viscosity coefficients globally.

Card

- Q1 = 1.5. Quadratic Artificial Viscosity.
- Q2 = 0.06. Linear Artificial Viscosity.
- TYPE = -2. Internal energy dissipated by the viscosity in the shell elements is computed and included in the overall energy balance.

*CONTROL_CONTACT

Specifies the defaults for computations of contact surfaces.

Card 1

- SLSFAC = 0 (uses the default = 0.1). Scale factor for sliding interface penalties.
- RWPNAL = 0. Scale factor for rigid wall penalties. When equal to 0 the constrain method is used and nodal points which belong to rigid bodies are not considered.
- ISLCHK = 1. Initial penetration check in contact surfaces. When set to 1 there is no checking.
- SHLTHK = 1 (default). Shell thickness considered in surface to surface and node to surface contact types. When set to 1, thickness is considered but rigid bodies are excluded.
- PENOPT = 1 (default). Penalty stiffness value option.
- THKCHG = 0 (default).

- ORIEN = 2. Automatic reorientation for contact segments during initialization. When set to 2 it is active for manual (segment) and automated (part) input.

- ENMASS = 0. This parameter regulates the treatment of the mass for eroded nodes in contact. When set to 0 eroding nodes are removed from the calculation.

Card 2

- USRSTR = 0. Storage per contact interface for user supplied interface control subroutine. When set to 0 no input data is read and no interface storage is permitted in the user subroutine.

- Default values are used for all other parameters.

Card3

- SFRIC = 0. Default static coefficient of friction.

- Default values are used for all other parameters.

Card4

- IGNORE = 2. Specifies whether to ignore initial penetrations in the *CONTACT_AUTOMATIC options. When set to 2 initial penetrations are allowed to exist by tracking them. Also, warning messages are printed with the original and the recommended coordinates of each contact node.

- FRCENG = 1. Calculate frictional energy in contact. Convert mechanical frictional energy to heat when doing a coupled thermal-mechanical problem.

- SKIPRWG = 0 (default).

- OUTSEG = 1. Yes, output each beam spot weld contact node and its target segment for *CONTACT_SPOTWELD into D3HSP file.

- SPOTSTP = 0 (default).

- SPOTDEL = 1.Yes, delete the attached spot weld element if the nodes of a spot weld beam or solid element are attached to a shell element that fails and the nodes are deleted.

- SPOTHIN = 0.5. This factor can be used to scale the thickness of parts within the vicinity of the spot weld. This factor helps avert premature weld failures due to contact of the welded parts with the weld itself. Should be greater than zero and less than one.

*CONTROL_ENERGY

Specifies the controls for energy dissipation options.

Card

- HGEN = 2. Hourglass energy is computed and included in the energy balance. Results are reported in ASCII files GLSTAT and MATSUM.

- RWEN = 2 (default).

- SLNTEN = 2. Sliding interface energy dissipation is computed and included in the energy balance. Results are reported in ASCII files GLSTAT and SLEOUT.
- RYLEN = 2. Rayleigh energy dissipation is computed and included in the energy balance. Results are reported in ASCII file GLSTAT.

*CONTROL_HOURGLASS

Defines the default values of the hourglass control type and coefficient. See *HOURGLASS (p. 159) for additional information.

Hourglass Controls	
Hourglass Type	Program Controlled
LS-DYNA ID	0
Default Hourglass Coefficient	0.1

Card

- IHQ = **Hourglass Type** from the Hourglass Controls section of the **Analysis Settings** (Defaults to 1, Standard LS-DYNA Hourglass):
  - = 1 if the Hourglass Type is set to Standard LS-DYNA.
  - = 2 if the Hourglass Type is set to Flanagan-Belytschko Viscous Form.
  - = 3 if the Hourglass Type is set to Exact Volume Flanagan-Belytschko Viscous Form.
  - = 4 if the Hourglass Type is set to Flanagan-Belytschko Stiffness Form.
  - = 5 if the Hourglass Type is set to Exact Volume Flanagan-Belytschko Stiffness Form.
  - = 6 if the Hourglass Type is set to Belytschko-Bindeman.
  - = 7 if the Hourglass Type is set to Belytschko-Bindeman Linear Total Strain.
  - = 8 if the Hourglass Type is set to Full Projection Warping Stiffness.
- QH = **Default Hourglass Coefficient** from the Hourglass Controls section of the Analysis Settings (Default to 0.1).

*CONTROL_IMPLICIT_SOLUTION

Controls the method used in implicit calculations.

- ILIMIT is read from **Iterations Per Reformation** under the Implicit Controls section of the **Analysis Settings**.
  - = 0 (default) when **Solution Method** under the Implicit Controls section of the **Analysis Settings** is set to BFGS Slightly Nonlinear.
  - = 6 when **Solution Method** under the Implicit Controls section of the **Analysis Settings** is set to BFGS Moderately Nonlinear.

- = 1 when **Solution Method** under the Implicit Controls section of the **Analysis Settings** is set to Full Newton.
- Editable when **Solution Method** under the Implicit Controls section of the **Analysis Settings** is set to BFGS Custom.

- MAXREF is read from **Maximum Reformation** under the Implicit Controls section of the **Analysis Settings**.
  - = 0 (default) when **Solution Method** under the Implicit Controls section of the **Analysis Settings** is set to BFGS Slightly Nonlinear
  - = 8 when **Solution Method** under the Implicit Controls section of the **Analysis Settings** is set to BFGS Moderately Nonlinear
  - = 30 when **Solution Method** under the Implicit Controls section of the **Analysis Settings** is set to Full Newton
  - < 0 (negative) when **Force Convergence** under the Implicit Controls section of the **Analysis Settings** is set to Yes
  - Editable when **Solution Method** under the Implicit Controls section of the **Analysis Settings** is set to Full Newton or BFGS Custom
- RCTOL is read from **Residual Relative** under the Implicit Controls section of the **Analysis Settings**.
  - = 0 (default) & editable
- ABSTOL is read from **Absolute Tolerance** under the Implicit Controls section of the **Analysis Settings**
  - = 0 (default) & editable

*CONTROL_IMPLICIT_SOLVER

Controls the linear equation solver.

- LCPACK
  - = 0 (default) when **Matrix assembly** under the Implicit Controls section of the **Analysis Settings** is set to Symmetric Linear Solver
  - = 3 when **Matrix assembly** under the Implicit Controls section of the **Analysis Settings** is set to Nonsymmetric Linear Solver

*CONTROL_IMPLICIT_GENERAL

Activates implicit analysis and defines associated control parameters.

- DT0 is read from **Initial Time Step** under the Implicit Controls section of the **Analysis Settings**.
  - = ENDTIM / 100 (default); ENDTIM is read from **End Time** in the Step Controls section of the **Analysis Settings**

*CONTROL_IMPLICIT_AUTO

Defines parameters for automatic time step control during implicit analysis.

- DTMAX is read from **Initial Time Step** under the Implicit Controls section of the **Analysis Settings**.
  - = DT0 * 10 (default); ENDTIM is read from **End Time** in the Step Controls section of the **Analysis Settings**

*CONTROL_MPP_DECOMPOSITION_DISTRIBUTE_ALE_ELEMENTS

Ensures ALE elements are evenly distributed to all processors. The input card (with no parameters) is added by default, for MPP calculations. The input card will not be written if the property **Distribute ALE Elements to All Processors** is set to No in the **ALE Controls** section of the Analysis Settings object.

*CONTROL_MPP_IO_LSTC_REDUCE

Use LST's own reduce routine to consistently sum floating point data among processors. This keyword is written when all of the following conditions are met:

- Processing Type under **CPU and Memory Management** is set to MPP
- Explicit Solution Only under **Solver Controls** is set to No
- Default Solver Controls Cards under **Advanced** is set to Keep

*CONTROL_MPP_IO_NOD3DUMP

Suppresses the output of the d3dump and runrsf files. This keyword is written when the following conditions are met:

- Processing Type under **CPU and Memory Management** is set to MPP
- Explicit Solution Only under **Solver Controls** is set to No
- Default Solver Controls Cards under **Advanced** is set to Keep
- Suppress Restart files for Distributed Calculations under **Output Controls** is set to Yes

*CONTROL_MPP_IO_NODUMP

Suppresses the output of all dump files and full deck restart files. This keyword is written when the following conditions are met:

- Processing Type under **CPU and Memory Management** is set to MPP
- Explicit Solution Only under **Solver Controls** is set to No
- Default Solver Controls Cards under **Advanced** is set to Keep
- Suppress Restart files for Distributed Calculations under **Output Controls** is set to Yes

*CONTROL_MPP_IO_NOFULL

Suppresses the output of the full deck restart files. This keyword is written when the following conditions are met:

- Processing Type under **CPU and Memory Management** is set to MPP
- Explicit Solution Only under **Solver Controls** is set to No
- Default Solver Controls Cards under **Advanced** is set to Keep
- Suppress Restart files for Distributed Calculations under **Output Controls** is set to Yes

*CONTROL_OUTPUT

This keyword controls the printing of various LS-DYNA output text files.

Card

- NPOPT is the only parameter that is set. The value is set to 1. With that parameter set, nodal coordinates, element connectivities, rigid wall definitions, nodal SPCs, initial velocities, initial strains,adaptive constraints, and SPR2/SPR3 constraints are not printed.

*CONTROL_PARALLEL

Controls parallel processing usage for shared memory computers by defining the number of processors and invoking the optional consistency of the global vector assembly.

Card

- CONST = 1. Consistency flag disabled for a faster solution

*CONTROL_SOLUTION

Specify the analysis solution procedure if thermal, coupled thermal analysis, or structural-only is performed.

Card

- SOLN
  - = 0. Structural analysis only, if the **Solver Type** is set to Program Controlled or Structural Analysis Only.
  - = 2. Coupled structural thermal analysis, if the **Solver Type** is set to Coupled Structural Thermal Analysis.

*CONTROL_SHELL

Specifies global parameters for shell element types.

Card

- WRPANG = 20 (default). Shell element warpage angle in degrees. If a warpage greater than this angle is found, a warning message is printed.
- ESORT = 1, full automatic sorting of triangular shell elements to treat degenerate quadrilateral shell elements as C^0 triangular shells.
- IRNXX = -1, shell normal update option. When set to -1, fiber directions are recomputed at each cycle.
- ISTUPD = 4, shell thickness update option for deformable shells. Membrane strains cause changes in thickness in 3 and 4 node shell elements, however elastic strains are neglected. This option is very important in sheet metal forming or whenever membrane stretching is important. For crash analysis, setting 4 may improve energy conservation and stability.
- THEORY = 2 (default). Belytschko-Tsay formulation.
- BWC = 1. For this setting, Belytschko-Wong-Chiang warping stiffness is added.
- MITER = 1 (default). Plane stress plasticity: iterative with 3 secant iterations.
- PROJ = 1, the full projection method is used for the warping stiffness in the Belytschko-Tsay and Belytschko-Wong-Chiang shell elements. This option is required for explicit calculations.
- NFAIL1 = 1. Flag to check for highly distorted under-integrated shell elements, print a message, and delete the element.
- NFAIL4 = 1. Flag to check for highly distorted fully-integrated shell elements, print a message, and delete the element.
- CNTO = 2. Flag to account for shell reference surface offsets in the contact treatment. Offsets are treated using the user defined contact thickness which may be different than the shell thickness used in the element.

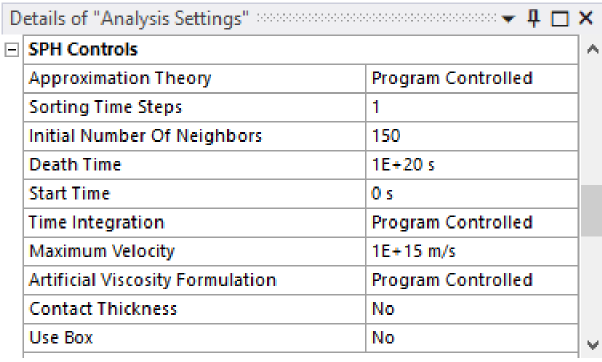
*CONTROL_SOLID

Specifies global parameters for solid element types.

Card

- ESORT = 1, full automatic sorting of tetrahedron and pentahedron elements to treat degeneracies. Degenerate tetrahedrons will be treated as ELFORM = 10 and pentahedron as ELFORM = 15 solids respectively (see *SECTION_SOLID).

*CONTROL_SPH



Provide controls relating to SPH solutions. Found in the Analysis Settings object Details panel.

Card

- FORM = **Approximation Theory** value from the **SPH Controls** section of the Details panel as indicated below. The formulation stands for the method used to discretize the momentum equation.
  - =0: If set to Default Formulation or Program Controlled
  - =1: If set to Renormalization Approximation
  - =5: If set to Fluid Particle Approximation
  - =6: If set to Fluid Particle With Renormalization Approximation
  - =7: If set to Total Lagrangian Formulation
  - =8: If set to Total Lagrangian Formulation With Renormalization
  - =12: If set to Moving Least Squares
  - =13: if set to ISPH
  - =15: If set to Enhanced Fluid Formulation
  - =16: If set to Enhanced Fluid Formulation With Renormalization

  Note:

  Moving Least Squares and ISPH are only available for an MPP simulation. You must set the processing type to MPP in the **CPU and Memory Management** section of the Details panel in order to use it.

- NCBS = **Sorting Time Steps** from the **SPH Controls** section of the Details panel.
- BOXID = ID of the box specified in **Box Name** in the **SPH Controls** section of the Details panel. Only Standard Box IDs are permitted in this field since SPH Boxes are selected automatically.

- MEMORY = **Initial Number of Neighbors** value from the **SPH Controls** section of the Details panel. Defines the initial number of neighbors per particle. This parameter is used to determine the initial memory allocation for the SPH arrays.
- DT = **Death Time** value from the **SPH Controls** section of the Details panel. Determines when the SPH calculations are stopped.
- START = **Start Time** . Particle approximations will be computed when time of the analysis has reached the value defined in START.
- DERIV = **Time Integration** value from the **SPH Controls** section of the Details panel. The smoothing length of each particle varies over time. Two types of equations can be used for the calculations.
  - =0: if set to Default or Program Controlled
  - =1: if set to Enhanced Energy Formulation
- MAXV = **Maximum Velocity** value from the **SPH Controls** section of the Details panel. Particles with a velocity greater than MAXV are deactivated. A negative MAXV will turn off the velocity checking.
- IAVIS = **Artificial Viscosity Formulation** value from the **SPH Controls** section of the Details panel. Defaults to 0.
  - =0: if set to Monaghan Formulation or Program Controlled
  - =1: Standard Formulation

*CONTROL_TERMINATION

Specifies the termination criteria for the solver.

Card

- ENDTIM = **End Time** in the **Step Controls** section of the **Analysis Settings** .
- ENDCYC = 10000000(constant) **Maximum Time Steps** .
- DTMIN = 0.001 (constant).
- ENDENG = 1000 (constant) **Maximum Energy Error** .
- ENDMAS = 100000 (constant) **Maximum Part Scaling** .

*CONTROL_THERMAL_TIMESTEP

This keyword is written if the simulation is determined to be a thermal one (for example, Coupled Structural Thermal Analysis). See Solver Type from the Solver Controls section of the **Analysis Settings**.

Thermal Step Controls	
Auto Time Stepping	Program Controlled
Initial Time Step	0.0
Minimum Time Step	0.0
Maximum Time Step	0.0
Time integration parameter	Crank-Nicholson Scheme
Time Integration	On

Card

- TS = **Auto Time Stepping** from Thermal Step Controls (default to 1):
  - 0 if **Auto Time Stepping** from Thermal Step Controls is set to No. The time step is fixed.
  - 1 if **Auto Time Stepping** from Thermal Step Controls is set to Yes. The time step is variable (may increase or decrease).
- TIP = **Time integration parameter** from Thermal Step Controls (default to Crank Nicholson Scheme TIP = 0):
  - 0 if **Time integration parameter** from Thermal Step Controls is set to Crank-Nicholson scheme.
  - 1 if **Time integration parameter** from Thermal Step Controls is set to Fully Implicit.
- ITS = **Initial Time Step** from Thermal Step Controls.
- TMIN = **Minimum Time Step** from Thermal Step Controls. If TMIN = 0.0, it is set to the structural explicit time step.
- TMAX = **Maximum Time Step** from Thermal Step Controls. If TMAX = 0.0, it is set to 100 * the structural explicit time step.

*CONTROL_THERMAL_SOLVER

This keyword is written if the simulation is determined to be a thermal one (for example, Coupled Structural Thermal Analysis). See Solver Type from the Solver Controls section of the **Analysis Settings** .

Thermal Solver Controls	
Thermal Analysis Type	Transient Analysis
Solver Type	Program Controlled
Fraction of Work Converted into Heat	1

Card

- ATYPE = **Thermal Analysis Type** from Thermal Solver Controls (default to 1)
  - 0 if **Thermal Analysis Type** from Thermal Solver Controls is set to Steady State Analysis.
  - 1 if **Thermal Analysis Type** from Thermal Solver Controls is set to Transient Analysis.

- SOLVER = **Solver Type** from Thermal Solver Controls (defaults to 1):
  - 1 if **Solver Type** is set to Symmetric Direct Solver.
  - 2 if **Solver Type** is set to Nonsymmetric Direct Solver.
  - 3 if **Solver Type** is set to Diagonal Scaled Conjugate Gradient Iterative.
  - 4 if **Solver Type** is set to Incomplete Choleski Conjugate Gradient Iterative.
  - 5 if **Solver Type** is set to Nonsymmetric Diagonal Scaled bi-Conjugate Gradient.
  - 12 if **Solver Type** is set to Diagonal Scaling Conjugate Gradient Iterative.
  - 13 if **Solver Type** is set to Symmetric Gauss-Siedel Conjugate Gradient Iterative.
  - 14 if **Solver Type** is set to SSOR Conjugate Gradient Iterative.
  - 15 if **Solver Type** is set to ILDLT0 Conjugate Gradient Iterative.
  - 16 if **Solver Type** is set to Modified ILDLT0 Conjugate Gradient Iterative.
- PTYPE = **Thermal problem type** from Thermal Nonlinear Controls.

Thermal Nonlinear Controls	
Thermal problem type	Nonlinear problem Gauss Point Temperature
Line Search	Program Controlled
Temperature Convergence	Program Controlled

  - 0 if **Thermal problem type** is set to Linear Problem.
  - 1 if **Thermal problem type** is set to Nonlinear problem Gauss Point Temperature.
  - 2 if **Thermal problem type** is set to Nonlinear problem Element Average Temperature.
- FWORK = **Fraction of Work Converted into Heat** from **Thermal Solver Controls** (defaults to 1).

*CONTROL_THERMAL_NONLINEAR

This keyword is written if the simulation is determined to be a thermal one (for example, Coupled Structural Thermal Analysis). See Solver Type from the Solver Controls section of the **Analysis Settings**.

Thermal Nonlinear Controls	
Thermal problem type	Nonlinear problem Gauss Point Temperature
Line Search	Program Controlled
Temperature Convergence	Program Controlled

Card

- THLSTL = **Line Search** from Thermal Nonlinear Controls (defaults to 0.0)
- TOL = **Temperature Convergence** from Thermal Nonlinear Controls (defaults to 0.0).

*CONTROL_TIMESTEP

Specifies conditions for determining the computational time step.

Step Controls	
End Time	1e-3
Time Step Safety Factor	0.9
Automatic Mass Scaling	Yes
Time Step Size	1E-07

Card

- DTINIT = 0 **Initial Time Step** .
- TSSFAC = **Time Step Safety Factor** from the **Step Controls** section of the **Analysis Settings** .
- ISDO = 0 (default). Basis of time size calculation for 4-node shell elements.
- TSLIMT = 0 **Minimum Element Timestep** ; the default value of 0.0 is used.
- DT2MS = the negative value of **Time Step Size** specified in the **Step Controls** section of the **Analysis Settings** , if **Automatic Mass Scaling** is set to **Yes** . If **Automatic Mass Scaling** is set to **No** the default value of 0.0 is used.
- LCTM = 0.
- ERODE = 0 for an SPG analysis. An SPG analysis is recognized by having at least one Section object with **Method** set to SPG. If it is not an SPG analysis, then ERODE = 1 (constant). This field can't be changed from the user interface. It is set based on the analysis conditions.
- MS1ST = 0 (default).

*DAMPING_GLOBAL

Specifies the mass weighted nodal damping applied globally to the nodes of deformable bodies and the center of mass of rigid bodies.

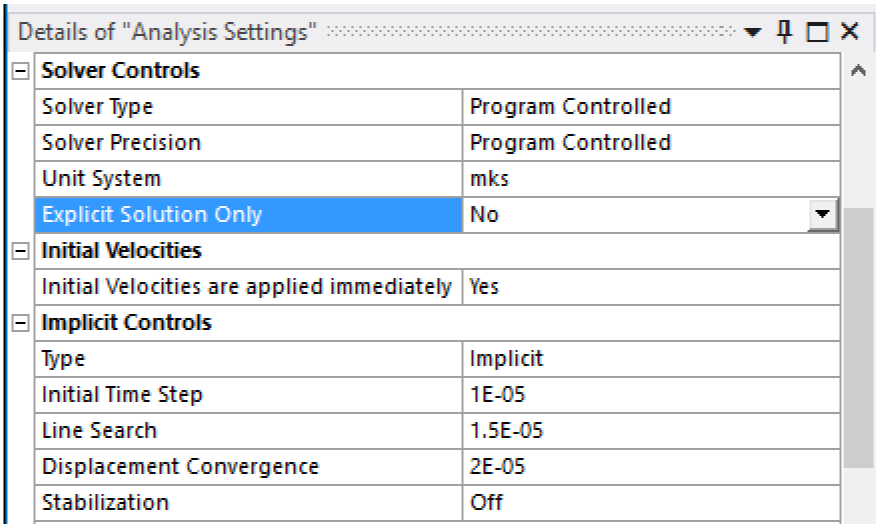
Damping Controls	
Global Damping	Yes
Magnitude	0.0

Card

- LCID = 0, a constant damping factor will be used as specified in VALDMP.
- VALDMP = **Magnitude** from the **Damping Controls** section of the **Analysis Settings** (defaults to zero if **Global Damping** is set to No).

*CONTROL_IMPLICIT_GENERAL

Setting **Explicit Solution Only** to No in the Solver Controls section of the **Analysis Settings** enables the use of the Implicit solver and displays the Implicit Controls section.



Card

- DT0 is read from **Initial Time Step** under the Implicit Controls section of the **Analysis Settings**
- IMFLAG = 1, to indicate an implicit analysis

*CONTROL_IMPLICIT_SOLUTION

Card

- DCTOL is read from **Displacement Convergence** under the Implicit Controls section of the Analysis Settings
- LSTOL is read from **Line Search** under the Implicit Controls section of the Analysis Settings

*CONTROL_IMPLICIT_SOLVER

Card

- LPRINT = 1 (Linear solver print flag, controls screen and message file output)

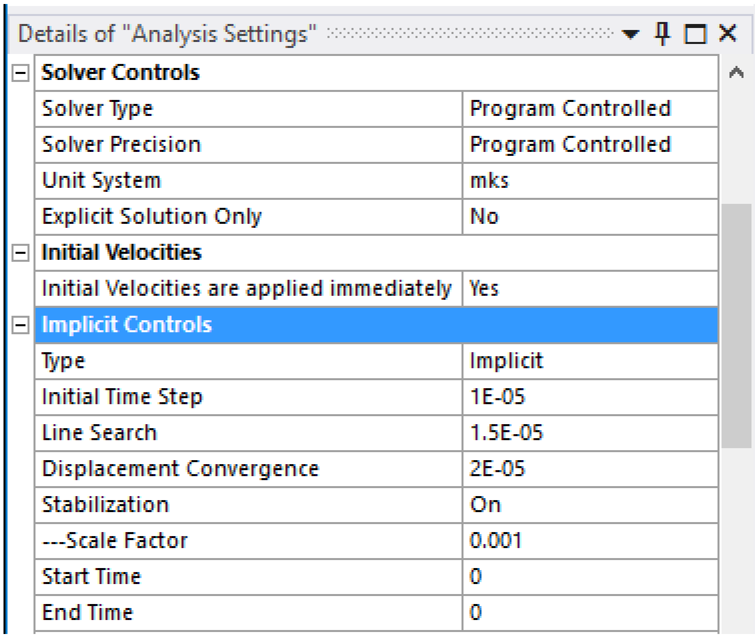
*CONTROL_IMPLICIT_AUTO

Card

- IAUTO = 1 (Automatic time step control flag)

*CONTROL_IMPLICIT_STABILIZATION

This keyword is written when **Stabilization** is set to On in the Implicit Controls section of the **Analysis Settings**



Details of "Analysis Settings"

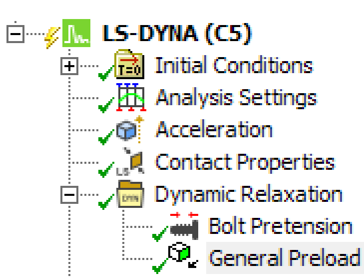
Solver Controls	
Solver Type	Program Controlled
Solver Precision	Program Controlled
Unit System	mks
Explicit Solution Only	No
Initial Velocities	
Initial Velocities are applied immediately	Yes
Implicit Controls	
Type	Implicit
Initial Time Step	1E-05
Line Search	1.5E-05
Displacement Convergence	2E-05
Stabilization	On
---Scale Factor	0.001
Start Time	0
End Time	0

Card

- IAS = 1 (to indicate the stabilization is active)
- SCALE is read from **–Scale Factor** in the Implicit Controls section of the **Analysis Settings**
- TSTART is read from **Start Time** in the Implicit Controls section of the **Analysis Settings**
- TEND is read from **End Time** in the Implicit Controls section of the **Analysis Settings**

5.4. Dynamic Relaxation Support

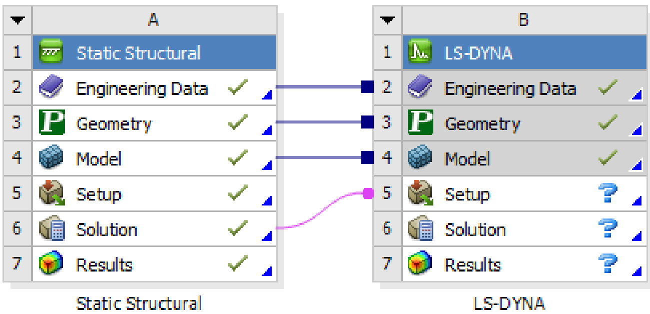
Sometimes a preload must be applied prior to an explicit dynamic analysis, especially for engineering problems where a structure's initial stress state affects its dynamic response. Dynamic relaxation (p. 56) serves that purpose. It can be achieved by using an implicit solver or the explicit solver itself.



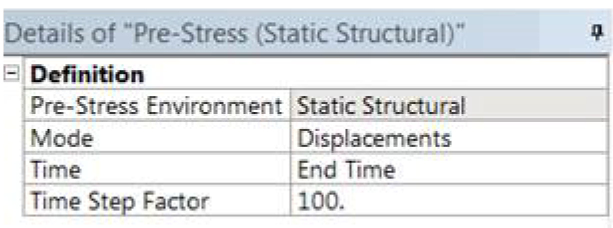
*CONTROL_DYNAMIC_RELAXATION

In LS-DYNA, three types of dynamic relaxation are available.

The first type of dynamic relaxation is "Explicit after Ansys Solution", which involves performing an Ansys implicit structural analysis followed by a LS-DYNA explicit dynamic analysis.



In this implicit-to-explicit sequential solution, you must first run an Ansys implicit structural analysis to apply a preload to the structure being analyzed. In this implicit analysis, completely constrain all of the nodes of any elements that will only be used in the explicit analysis (for example, the bird in a bird-strike problem). The nodal displacements and rotations at the time specified in the pre-stress environment from the Ansys implicit solution are written to the LS-DYNA dynamic relaxation file `input.sif` .

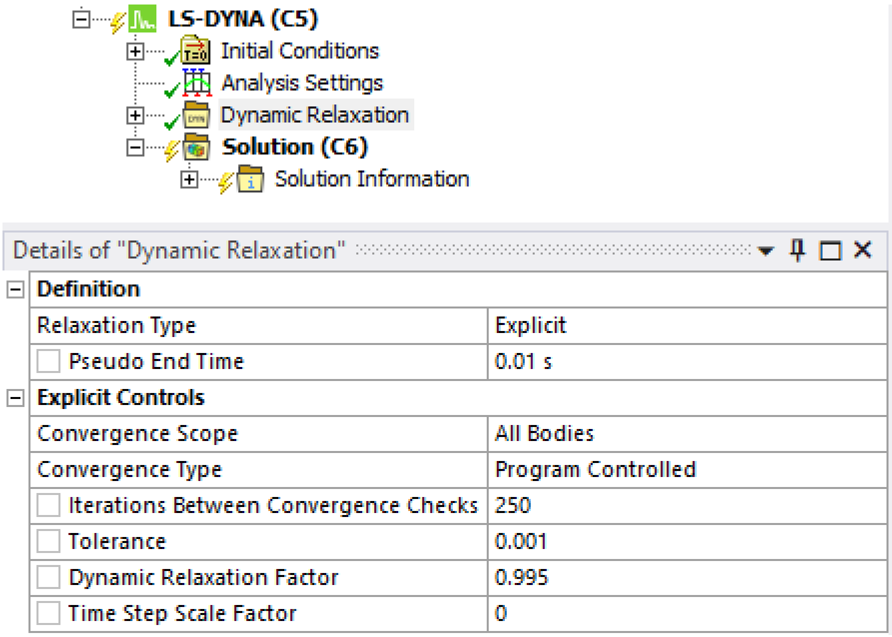


Details of "Pre-Stress (Static Structural)"

Definition	
Pre-Stress Environment	Static Structural
Mode	Displacements
Time	End Time
Time Step Factor	100.

Card

- TSSFDR is the time scale factor for computed time steps during dynamic relaxation, read from the **Time Step Scale Factor** under the Explicit Controls section. If zero, the value is set to TSSFAC (defined through the **Time Step Safety Factor** under **Analysis Settings**). After converging, the scale factor is reset to TSSFAC.



- NRCYCK is the number of iterations between convergence checks for dynamic relaxation option, read from **Iterations Between Convergence Check** under the Explicit Controls section. Default = 250.

- DRTOL is the convergence tolerance for the dynamic relaxation option, read from **Tolerance** under the Explicit Controls section. Default = 0.001.

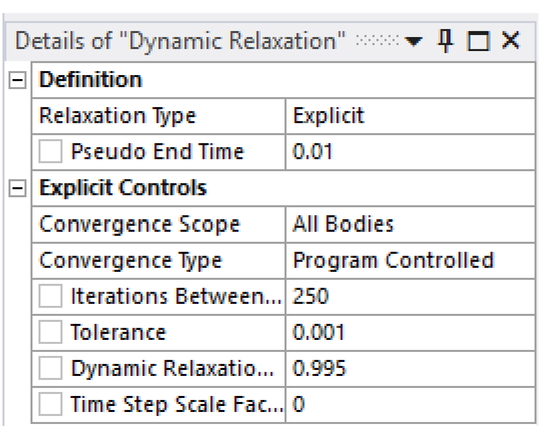
- DFFCTR is the dynamic relaxation factor read from **Dynamic Relaxation Factor** under the Explicit Controls section. Default = 0.995.

- IDRFLG is the dynamic relaxation flag for stress initialization and is set to 2 (initialization to a prescribed geometry).

The second type of dynamic relaxation, Explicit, allows the explicit solver to conduct a quasi-static analysis by increasing the damping until the kinetic energy drops to zero.

In that case, the solver will check the kinetic energy every 250 cycles (by default) until the kinetic energy from the applied preload is dissipated.

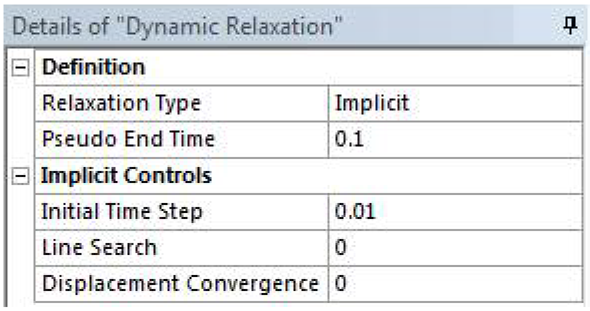
Five preloads are supported: Gravity, Acceleration, Rotational Acceleration, Rotational Velocity, and Bolt Pretension.



The dynamic relaxation parameters are identical to the dynamic relaxation of type Explicit after Ansys Solution, except for the parameter IDRFLG (dynamic relaxation flag for stress initialization):

- IDRFLG = 1 (dynamic relaxation is activated) if the **Convergence Scope** is set to All Bodies
- IDRFLG = 3 (dynamic relaxation is activated), if the **Convergence Scope** is set to Geometry Selection.

The third type of dynamic relaxation allows LS-DYNA to initialize Implicitly instead of Explicitly.



The dynamic relaxation parameters are identical to the dynamic relaxation of type Explicit after Ansys Solution, except for the parameter IDRFLG (dynamic relaxation flag for stress initialization):

- IDRFLG = 5 (initialize implicitly).

The following 4 keywords setting up the implicit initialization are then written:

*CONTROL_IMPLICIT_GENERAL_DYN

- DT0 is read from **Initial Time Step** under the Implicit Controls section

*CONTROL_IMPLICIT_SOLUTION_DYN

- DCTOL is read from **Displacement Convergence** under the Implicit Controls section
- LSTOL is read from **Line Search** under the Implicit Controls section

*CONTROL_IMPLICIT_SOLVER_DYN

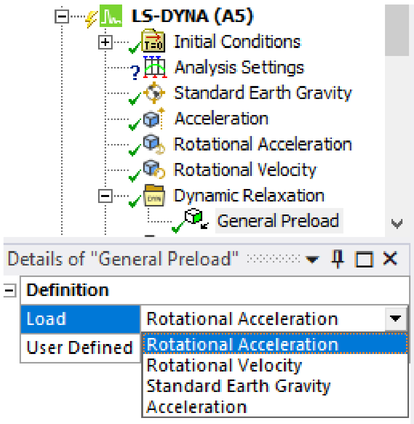
- LPRINT = 1 (Linear solver print flag, controls screen and message file output)

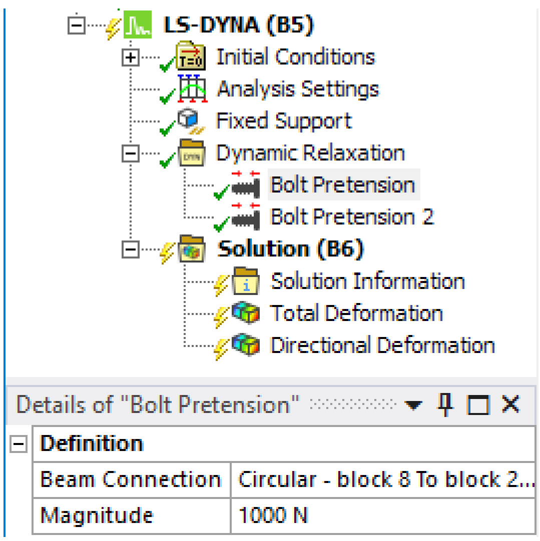
*CONTROL_IMPLICIT_AUTO_DYN

- IAUTO = 1 (Automatic time step control flag)

5.4.1. Available Preloads for Dynamic Relaxation

Preloading can be done with a General Preload or Bolt Pretension (p. 54) object. Four loads are available for General Preload: Standard Earth Gravity, Acceleration, Rotational Acceleration, and Rotational Velocity. The **Load** field must be assigned accordingly.





*INITIAL_AXIAL_FORCE_BEAM

This keyword is used when a given beam connection has a Bolt Pretension load applied in the dynamic relaxation.

5.5. ALE Support

*ALE_MULTI-MATERIAL_GROUP

The *ALE_MULTI-MATERIAL_GROUP keyword is automatically added with a separate multi-material group for each body defined with a reference frame set to S-ALE Domain or S-ALE Fill, or included in a section definition with element formulation type 11 (1 point ALE Multi-Material Element).

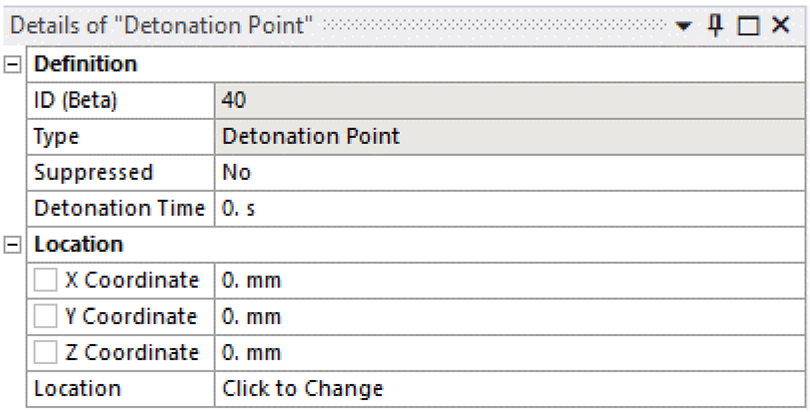
*ALE_STRUCTURED_MESH

*ALE_STRUCTURED_MESH_CONTROL_POINTS

The *ALE_STRUCTURED_MESH and associated *ALE_STRUCTURE_MESH_CONTROL_POINTS keywords are added for each body with the S-ALE Domain reference frame. Refer to ALE Workflow (p. 79) for further details.

*INITIAL_DETONATION

An *INITIAL_DETONATION keyword is added for each detonation point in the analysis. Detonation points are applied to all high explosive materials by default.



Card 1

- X = X Coordinate field of the Detonation Point
- Y = Y Coordinate field of the Detonation Point
- Z = Z Coordinate field of the Detonation Point
- LT = Detonation Time field of the Detonation Point

*INITIAL_VOLUME_FRACTION_GEOMETRY

The *INITIAL_VOLUME_FRACTION_GEOMETRY keyword is added when the model contains S-ALE bodies (that is, those with Reference Frame S-ALE Domain or S-ALE Fill). The S-ALE Domain bodies are used to fill the Background ALE Mesh card and each S-ALE Fill body is added as Container card. If the S-ALE Fill bodies have a Velocity initial condition, then the details are included in the Container card.

*ALE_VOID_PART

Added when a Section is defined with element formulation 12.

*ALE_ESSENTIAL_BOUNDARY

Added when the ALE Boundary object is part of the analysis.

*ALE_STRUCTURED_FSI / *CONSTRAINED_LAGRANGE_IN_SOLID

Added when a Coupling object is defined under the Connections branch. The keyword used depends on the Fluid Structure Interaction Type chosen. See Section Coupling (p. 28) for further details.

5.6. Part Setup

*PART

Defines geometry bodies.

Card 1

- HEADING = name of the body specified in the Mechanical environment.

Card 2

- PID = ID of the part. It is set in the LS-DYNA solver and does not reflect the ID specified in the mesh definition of the model.
- SECID = ID of the section keyword associated with the part (see *SECTION).
- MID = ID of the material keyword associated with the part (see *MAT).
- EOSID = ID of the equation of state associated with the material of this part (*EOS and *MAT). If there is no EOS keyword associated with this part then this parameter is set to 0.
- HGID = ID of the hourglass keyword associated with the part (see *HOURGLASS (p. 159)). If there is no hourglass keyword associated with this part then this parameter is set to 0.

Note:

When you insert a PART object using the Keyword Manager, setting the field **Material defined in *Mat section** to Program Controlled specifies that the solver use the material defined for the Mechanical part that the keyword object is scoped to. Alternatively, you have the option to choose from other materials that are defined in the system.

---

*SECTION_BEAM

Defines cross sectional properties for beam, truss, spot weld and cable elements.

Card 1

- SECID = ID of the section.
- ELFORM = 1 or 2 (default). ELFORM = 2 is set for user defined cross sections. The default element formulation option can be changed using the **Section** object found under **Part** on the LSDYNA Pre tab of LS-DYNA.
- SHRF = 0.833 (default).

- QR = 0, which LS-DYNA defaults to 2, quadrature rule is 2x2 Gauss. If the cross sectional area of the beam is complex or user-defined, this parameter becomes IRID and is assigned the negative value of the IRID parameter in the corresponding *INTEGRATION_BEAM keyword (see above for details).

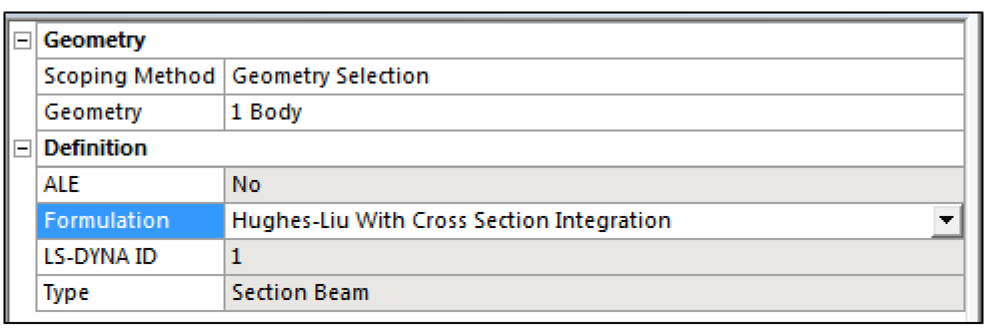
- CST = 2 for solid cross sections nand hollow cross sections (arbitrary user defined integration rule).

Card 2

- for solid types or hollow cylinders
  - TS1 = width of beam. This refers specifically to the dimension at node 1.
  - TS2 = TS1. This refers specifically to the dimension at node 2.
  - TT1 = 1. Height of beam. This refers specifically to the dimension at node 1. Set to zero for circular solids.
  - TT2 = TT1. This refers specifically to the dimension at node 2. Set to zero for circular solids. These parameters are overwritten by the *INTEGRATION_BEAM defined for these types.
- for general symmetric types
  - A = cross-sectional area.
  - ISS = Iyy, moment of inertia about the local s-axis.
  - ITT = Izz, moment of inertia about the local t-axis.
  - IST = Iyz.
  - J = Ixx.

The **Section** object found under **Part** on the LSDYNA Pre tab of LS-DYNA allows you to modify the default generated values .

In presence of line Bodies:



- ELFORM = **LS-DYNA ID** from the **Definition** section of the **Section** object. This field is read only. The actual value of the element formulation is set by the **Formulation** section of the **Section** object.

If the formulation is one of the following, the Card is calculated similarly to the above definition for *SECTION_BEAM, and an *INTEGRATION_BEAM is written:

- Hughes -Liu with cross section integration

- Integrated warped beam
- Belytschko Schwer full cross-section integration
- Belytschko Schwer tubular beam with cross-section integration

If the formulation is one of the following:

- Belytschko Schwer resultant beam (resultant)
- Truss (resultant)
- Belytschko Schwer full cross-section integration

Card 2 is modified and uses the syntax for the alternative form for formulations 2, 3, and 12.

- STYPE is calculated from the section type defined in Ansys DesignModeler or Discovery Modeling.
- D1 - D6 are calculated from the dimensions defined in Ansys DesignModeler.

If the formulation is Discrete/ Beam Cable, an additional panel is available:

Figure 5.1: Discrete and Cable Controls when the Option is set to Discrete Beam

Discrete and Cable Controls	
Option	Discrete Beam
Coordinate System	Global Coordinate System
Volume	150 [mm mm mm]
Mass Moment of Inertia	0.34 [tonne mm mm]
Longitudinal Stiffness X	100000 [N mm^-1]
Longitudinal Stiffness Y	110000 [N mm^-1]
Longitudinal Stiffness Z	120000 [N mm^-1]
Torsional Stiffness X	100000000 [N mm degree^-1]
Torsional Stiffness Y	200000000 [N mm degree^-1]
Torsional Stiffness Z	300000000 [N mm degree^-1]

A material Card (*MAT_LINEAR_ELASTIC_DISCRETE_BEAM (p. 153)) is added to allow definition of properties for the discrete Beam.

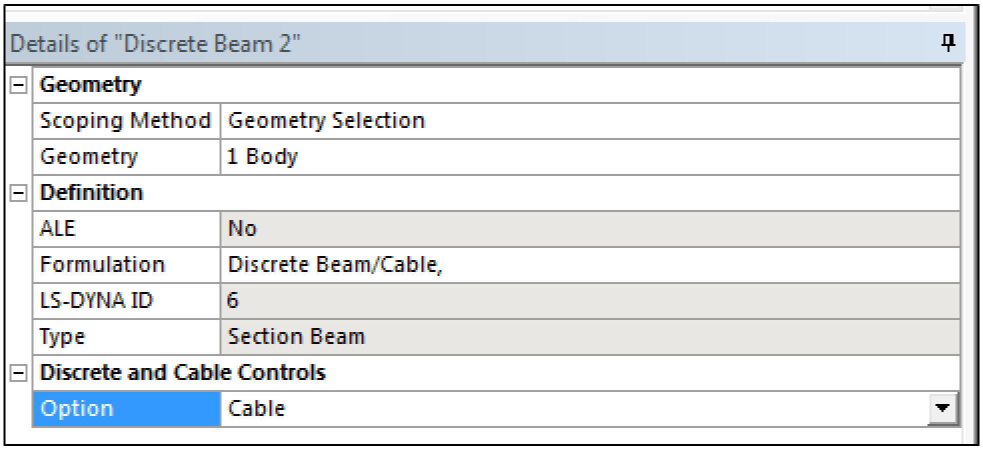
*MAT_LINEAR_ELASTIC_DISCRETE_BEAM

This card replaces the material defined in Engineering and its properties are calculated from it and the above panel.

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = Density of the Material
- TKR = **Longitudinal Stiffness X** from Discrete and Cable Controls
- TKS = **Longitudinal Stiffness Y** from Discrete and Cable Controls
- TKT = **Longitudinal Stiffness Z** from Discrete and Cable Controls

- RKR = **Torsional Stiffness X** from Discrete and Cable Controls
- RKS = **Torsional Stiffness X** from Discrete and Cable Controls
- RKT = **Torsional Stiffness X** from Discrete and Cable Controls

Figure 5.2: Discrete and Cable Controls when the Option is set to Cable



*MAT_CABLE_DISCRETE_BEAM

This card replaces the material defined in Engineering Data and its properties are calculated from it, and the above panel.

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = Density of the Material
- E = Young Modulus of the Material

A material Card (*MAT_CABLE_DISCRETE_BEAM) is added to allow definition of properties for the discrete Beam.

*SECTION_SHELL

Defines section properties for shell elements.

Card1

- SECID = ID of the section.
- ELFORM = 2 (default).
- SHRF = 0.8333 (default).
- NIP = 3 (default).

Card2

- T1 = thickness of body.

- T2-T4 = T1, shell thickness at nodes 2, 3 and 4.

The **Section** object found under **Part** on the LSDYNA Pre tab of LS-DYNA allows you to modify the default generated values .

In the presence of Surface Bodies:

Details of "Section"	
Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
ALE	No
Formulation	Fast (Co-Rotational) Hughes-Liu
LS-DYNA ID	11
Type	Section Shell
Through Thickness Integration Points	5 Point

- ELFORM = **LS-DYNA ID** from the **Definition** section of the **Section** object. This field is read only. The actual value of the element formulation is set by the **Formulation** section of the **Section** object.

- NIP = **Through Thickness Integration Points** from the **Definition** section of the **Section** object.

*SECTION_SOLID

Defines section properties for solid elements.

Card

- SECID = ID of the section.

- ELFORM =

  - 1 (default). Also, used for first-order hexahedral elements, 5-noded pyramids, 6-noded wedges or bodies with mixed element types that include tetrahedrons together with hexahedrons, pyramids, or wedges.

  - 10 if elements are first-order tetrahedrons.

  - 16 if the elements are second-order tetrahedrons.

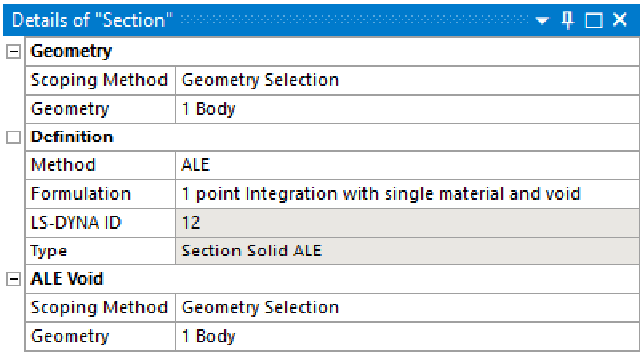
The **Section** object found under **Part** on the LSDYNA Pre tab of LS-DYNA allows you to modify the default generated values.

The **Method** field determines the type of Section_Solid keyword that is written. A value of FEM indicates finite element formulations are available.

Details of "Section"	
Geometry	
Scoping Method	Geometry Selection
Geometry	1 Body
Definition	
Method	FEM
Formulation	Constant Stress Solid Element
LS-DYNA ID	1
Type	Section Solid

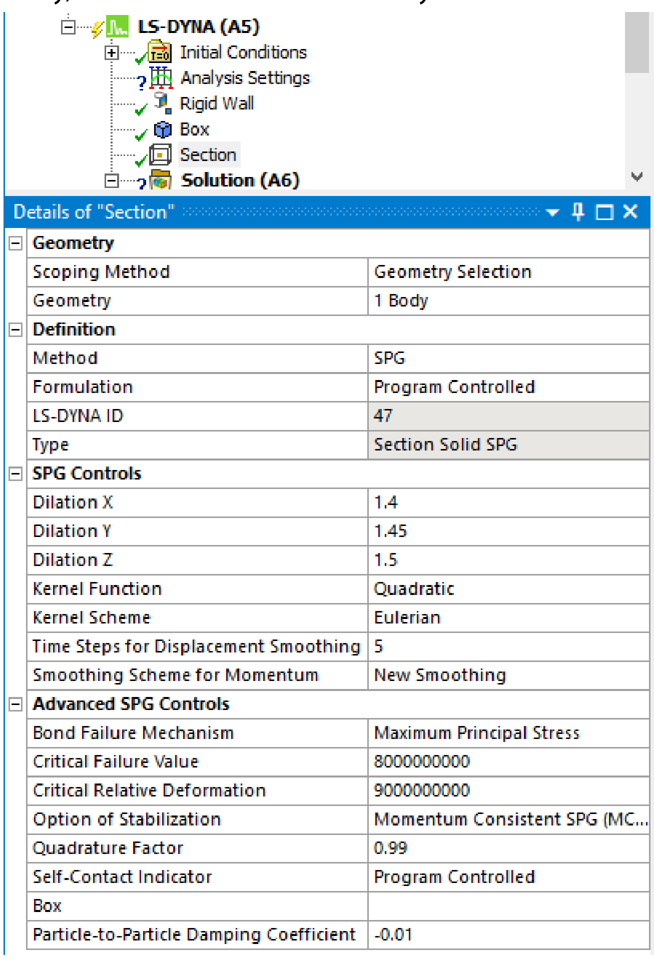
- ELFORM = **LS-DYNA ID** from the **Definition** section of the **Section** object. This field is read only. The actual value of the element formulation is set by the **Formulation** section of the **Section** object.

When **Method** is set to ALE, then the ALE specific formulations are available for selection. If the single material and void formulation is selected, then an additional geometry selection field is available to select the void bodies. The *INITIAL_VOID_PART card is created for these bodies.



*SECTION_SOLID_SPG

When the general setting Method is set to SPG, fields are provided to set section properties for an SPG body, in addition to the Geometry and Definition categories defined for all Section Solid objects.



SPG Controls

DX, DY, DZ: Corresponds to the **Dilation X, Y, Z** values entered.

ISPLINE: Set by the **Kernel Function** field.

Kernel: Set by the **Kernel Scheme** field.

SMSTEP: Set from the **Time Steps for Displacement Smoothing** field.

MSC: Set from the **Smoothing Scheme for Momentum** field. The option New Smoothing can be set only if a feature flag called Enable Momentum Consistent SPG in Section Object is enabled (this is a change in behavior from the previous release).

Advanced SPG Controls

IDAM: Set from the **Bond Failure Mechanism** field.

FS: Set from the **Critical Failure Value** field.

STRETCH: Set from the **Critical Relative Deformation** field.

ITB: Set from the **Option of Stabilization** field. The option Momentum Consistent SPG (MCSPG) Formulation can be set only if a feature flag called Enable Momentum Consistent SPG in Section Object is enabled (this is a change in behavior from the previous release).

MSFAC: Set from the **Quadrature Factor** field.

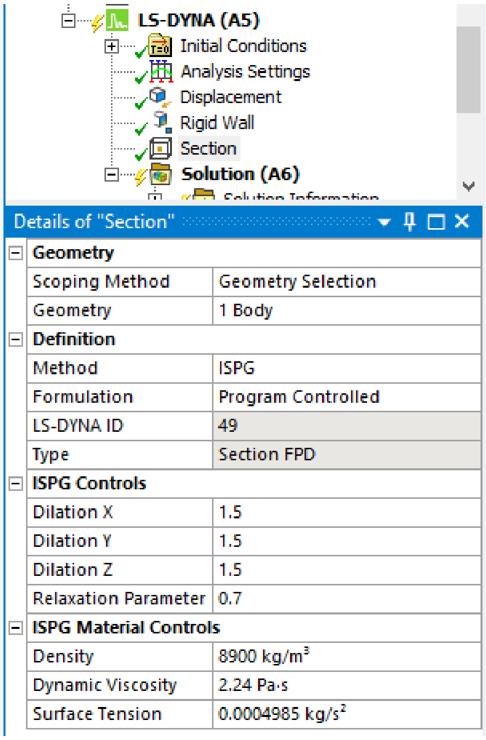
ISC: Set from the **Self-Contact Indicator** field.

BOXID: Set from the **Box** field.

PDAMP: Set from the **Particle-to-Particle Damping Coefficient** field. This option is visible only if a feature flag called Enable Momentum Consistent SPG in Section Object is enabled (this is a change in behavior from the previous release).

To use a box, you must first add a Bounding Box object (p. 60) and then select that object in the **Box** field.

*SECTION_SOLID_FPD, *MAT_IFPD



When **Method** is set to ISPG, categories are provided to set section properties for an ISPG body in addition to the Geometry and Definition categories defined for all Section Solid objects.

ISPG Controls

DX, DY, DZ: Corresponds to the Dilation X, Y, Z values entered

MCVISC: Set from the Relaxation Parameter field

ISPG Material Controls

The properties in this section correspond to fields in the *Mat_IFPD keyword.

RHO: Set from the Density field

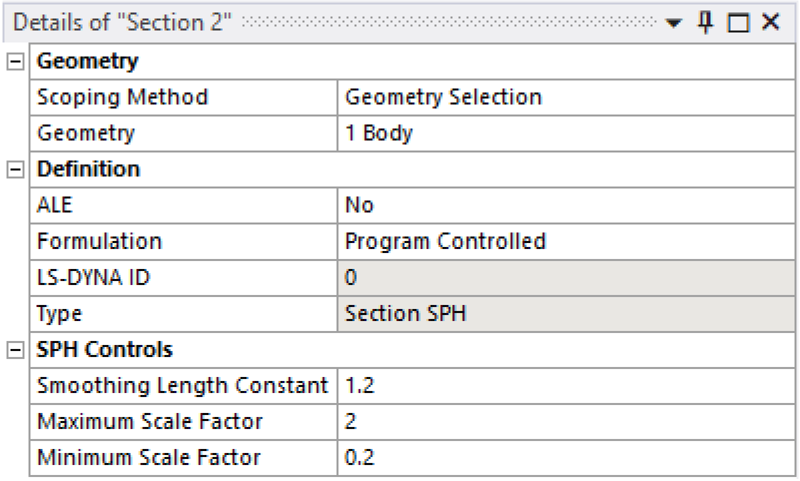
MU: Set from the Dynamic Viscosity field

GAMMA: Set from the Surface Tension field

*SECTION_SPH

Defines section properties for SPH particles.

If you insert a Section object from the LSDYNA Pre tab and scope the section to a particle body, the SPH Controls section appears in the Details panel and includes 3 parameters as shown in the figure below:



Card

- SECID: Section ID. SECID is referenced on the *PART card.
- CSLH = **Smoothing Length Constant** from the Details panel: This constant is used to compute the initial smoothing length of the particles. It is defined as the ratio between the initial smoothing length and the spacing between particles. Values between 1.05 and 1.3 are acceptable. Using a value less than 1 is prohibited. Values larger than 1.3 will increase the computation time. The default value is 1.2.

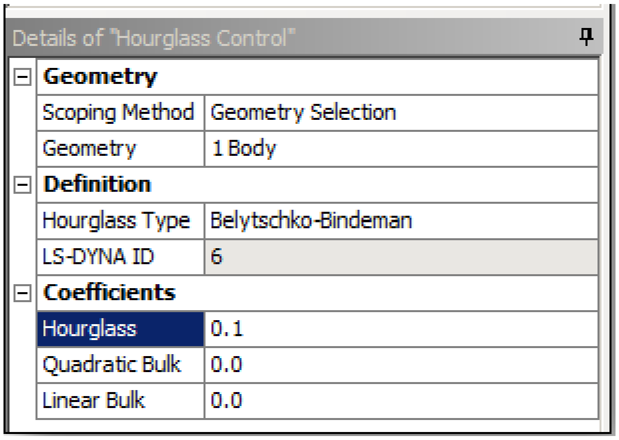
> **Note:**
>
> The smoothing length h(t) can vary over time with respect to the following inequality:
>
> CSLH * H_{min} < h(t) < CSLH * H_{max}
>
> To have a constant smoothing length over time, set $H_{min} = H_{max} = 1$.

- HMIN = **Minimum Scale Factor** from the Details panel: The maximum scale factor for the smoothing length.
- HMAX = **Maximum Scale Factor** from the Details panel:
- SPHINI: Optional initial smoothing length. (Default is used.)

*HOURGLASS

Defines hourglass and bulk viscosity properties that are referenced in the *PART keyword using the HGID parameter (see the *PART (p. 151) keyword). The Hourglass Control object can be used to set the SPH part viscosity. You must scope an SPH body to the object and enter the **Quadratic Bulk** viscosity and the **Linear Bulk** viscosity.

This keyword can be written using the **Hourglass Control** object found under **Part** on the LSDYNA Pre tab of an LS-DYNA analysis, which allows you to specify body scoped hourglass definition.



- HGID = ID of the part. It is set in the LS-DYNA solver and does not reflect the ID specified in the mesh definition of the model.
- IHQ = Hourglass Control Type.
  - = 1 if the Hourglass Type is set to Standard LS-DYNA.
  - = 2 if the Hourglass Type is set to Flanagan-Belytschko Viscous Form.
  - = 3 if the Hourglass Type is set to Exact Volume Flanagan-Belytschko Viscous Form.
  - = 4 if the Hourglass Type is set to Flanagan-Belytschko Stiffness Form.
  - = 5 if the Hourglass Type is set to Exact Volume Flanagan-Belytschko Stiffness Form.
  - = 6 if the Hourglass Type is set to Belytschko-Bindeman.
  - = 7 if the Hourglass Type is set to Belytschko-Bindeman Linear Total Strain.
  - = 8 if the Hourglass Type is set to Full Projection Warping Stiffness
- QM = **Hourglass** from the **Coefficients** section of the **Hourglass Control** object.
- Q1 = **Quadratic Bulk** from the **Coefficients** section of the **Hourglass Control** object.
- Q2 = **Linear Bulk** from the **Coefficients** section of the **Hourglass Control** object.
- IBQ = 1 Standard LS-DYNA Bulk Viscosity

You can define the default values of the hourglass control type and coefficient from the Hourglass Controls section of the **Analysis Settings** (see *CONTROL_HOURGLASS (p. 134)).

This keyword can also be created using the **Keyword Snippet** (see also, Commands objects) for the LS-DYNA solver. To use it, insert a **Keyword Snippet** under a Geometry body in the Tree Outline. The program will automatically substitute the HGID parameter in accordance with the *PART keyword of the associated body. All other parameters in the **Keyword Snippet** are transcribed literally.

If the keyword is entered in a **Keyword Snippet** anywhere else in the Tree Outline, it will be exported literally. This practice is not recommended, however, and a warning is provided in the header of **Keyword Snippet** objects when detected.

*CONSTRAINED_LAGRANGE_IN_SOLID

This keyword is used for reinforcements body interactions.

- The first keyword paramter is set to the ID of the component containing line bodies.
- The second keyword parameter is set to the ID of the component containing solid bodies.

This keyword is also used by the Coupling (p. 28) object.

*CONSTRAINED_RIGID_BODIES

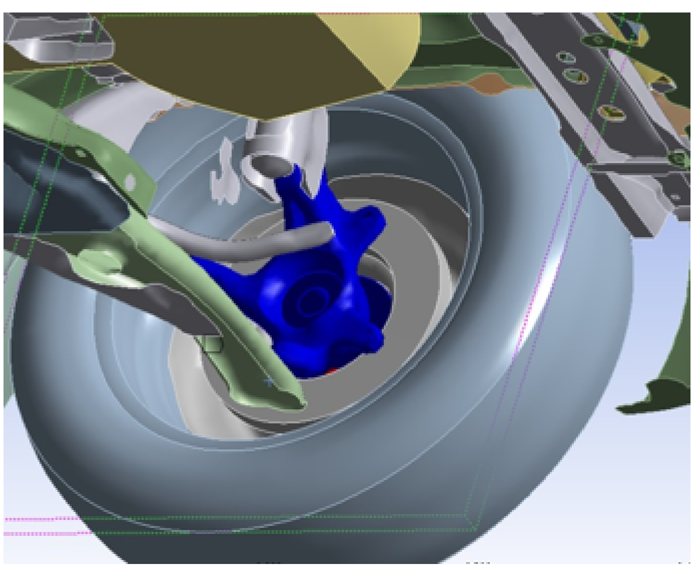
Specifies rigid bodies to be merged into one part. The resulting Part ID matches the ID of the rigid body designated as the target.

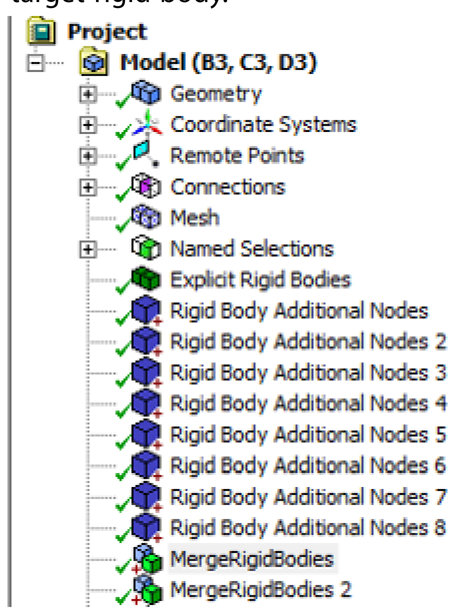
By constraining the rigid bodies together using a single multibody part you avoid specifying conflicting motion on the nodes shared among the rigid bodies. All boundary conditions applied to the target body will also be applied to all the contact bodies as well. Any boundary conditions that were applied to the contacts will be ignored.

Card

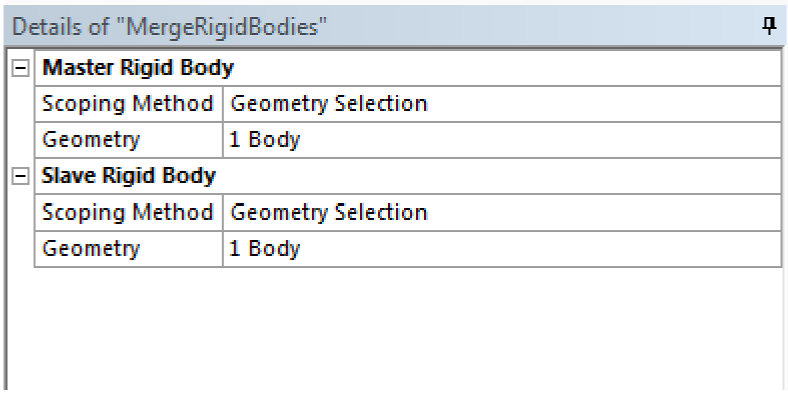
- PIDM = ID of the target rigid body.
- PIDS = ID of the contact rigid body.

The object **Master Rigid Body** allows you to specify the target rigid body.



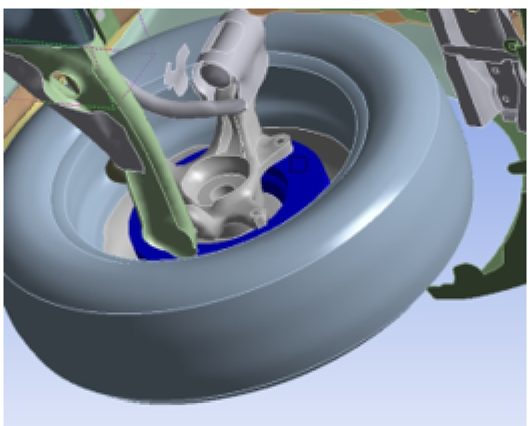


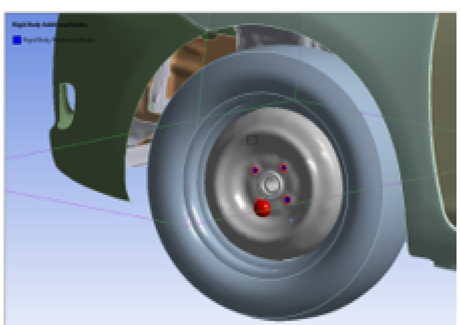
- PID = the part ID of the rigid body.
- NID = the set ID of additional nodes.

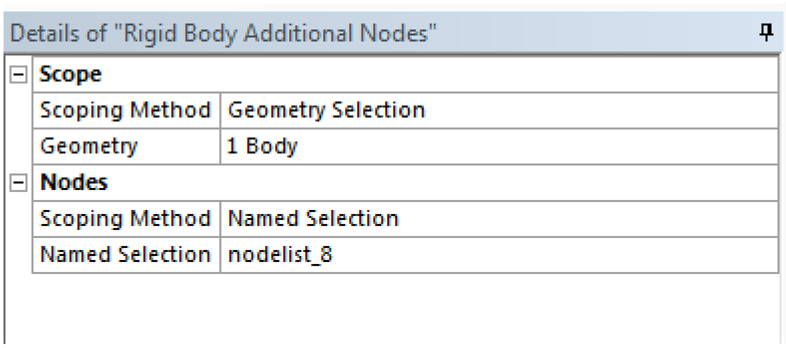


* CONSTRAINED_EXTRA_NODE_SET

The extra node set constraint type allows the addition of nodes (via a nodal component) to an existing rigid body. The nodal component that is added must not be attached to any other rigid body. The extra nodes that are added to a rigid body may be located anywhere in the model and may have coordinates outside those of the original rigid body. This option has many potential applications, including placing nodes where joints will be attached between rigid bodies, defining nodes where point loads will be applied, and defining a lumped mass at a specific location.







The underlying selected nodes will be part of the selected rigid body.

Card

- PID = ID of the target rigid body.
- NSID = ID of the set containing the selected nodes.

5.7. Engineering Data Materials and Equations of State

Equation of State (EOS) Keywords

The following are descriptions for *EOS keywords natively supported by LS-DYNA. More generally, any *EOS keyword may be introduced into the export file with the help of Commands objects in the Mechanical application (termed **Keyword Snippet** when referring to the LS-DYNA solver). To use it, insert a **Keyword Snippet** under a Geometry body in the Tree Outline. The program will automatically substitute the EOSID parameter, in accordance with the *PART keyword of the associated body. All other parameters in the **Keyword Snippet** are transcribed literally, overriding any values that would otherwise derive from the Engineering Data workspace.

If the *EOS keyword is entered in a **Keyword Snippet** anywhere else in the Tree Outline, it will be exported literally and the Engineering Data EOS information will also be exported, if present. This practice is not recommended, however, and a warning is provided in the header of **Keyword Snippet** objects when detected.

*EOS_GRUNEISEN

Specifies a shock equation of state. This keyword is created when a Shock EOS linear equation of state is present in the properties of a material that is used in the simulation and the Johnson Cook plasticity model is also present. The bilinear version of this equation of state is not currently supported.

Card 1

- EOSID = ID of the keyword, must be unique between the *EOS keywords.
- C = parameter C1 for a Linear Shock EOS property.
- S1 = parameter S1 for a Linear Shock EOS property.

- S2 = Parameter Quadratic S2 for a Linear Shock EOS property.
- S3 = 0.
- GAMAO = Gruneisen Coefficient for a Linear Shock EOS property.
- A = 0.

Card 2 - mandatory, left blank.

*EOS_LINEAR_POLYNOMIAL

Specifies the coefficients for a linear polynomial elastic EOS. The *EOS_LINEAR_POLYNOMIAL keyword is only created when the Johnson Cook strength property is added to the material model (which requires an EOS), but no other EOS has been specified. It is not directly available from the Engineering Data workspace, however.

Card 1

- EOSID = ID of the keyword, must be unique between the *EOS keywords.
- C0 = 0.
- C1 = Parameter A1.
- C2 = Parameter A2.
- C3 = Parameter A3.
- C4 = Parameter A4.
- C5 = Parameter A5.
- C6 = Parameter A6.

Card 2 - mandatory, left blank.

*EOS_IDEAL_GAS

This keyword is created when an Ideal Gas EOS is present in the properties of a material that is used in the simulation. For the LS-DYNA system in Mechanical, the parameters **Pressure Shift** and **Reference Temperature** are not passed to the solver. Other fields are passed to the solver as indicated below. Density is used by the associated *MAT_NULL (p. 177) keyword.

Properties of Outline Row 3: Air(Atmospheric)

	A	B	C	D	E
1	Property	Value	Unit		
2	Material Field Variables	Table			
3	Density	1.225	kg m^-3		
4	Specific Heat Constant Volume, C_v	717.6	J kg^-1 C^-1		
5	Ideal Gas EOS				
6	Adiabatic Exponent γ	1.4			
7	Adiabatic Constant	0			
8	Pressure Shift	0	Pa		
9	Reference Temperature	15.05	C		
10	Specific Internal Energy	2E+05	J kg^-1		
11	Initial Relative Volume, V0	0			

Card 1

- EOSID = ID of the keyword, must be unique between the *EOS keywords.
- CV0 = Specific Heat Constant Volume, C_v.
- CP0 = (Specific Heat Constant Volume C_v) × (Adiabatic Exponent).
- CL = 0.
- CQ = 0.
- T0 = (Specific Internal Energy) ÷ (Specific Heat Constant Volume, C_v).
- V0 = Initial Relative Volume, V0.
- VC0 = 0.

Card 2

- ADIAB =
  - 0: If Adiabatic Constant = 0.
  - 1: If Adiabatic Constant ≠ 0.

*EOS_JWL

For the LS-DYNA system in Mechanical, the parameters **Adiabatic Constant**, **Additional specific energy / unit mass**, **Begin Time**, and **End Time** are not passed to the solver. Other fields are passed to the solver as indicated below. Some of these fields are used by the associated *MAT_HIGH_EXPLOSIVE_BURN (p. 171) keyword.

Properties of Outline Row 6: C4

	A	B	C	D	E
1	Property	Value	Unit		
2	Material Field Variables	Table			
3	Density	1601	kg m^-3		
4	Explosive JWL				
5	Parameter A	6.0977E+11	Pa		
6	Parameter B	1.295E+10	Pa		
7	Parameter R1	4.5			
8	Parameter R2	1.4			
9	Parameter W	0.25			
10	Initial Relative Volume, V0	0			
11	C-J Detonation Velocity	8193	m s^-1		
12	C-J Energy / unit mass	5.621E+06	J kg^-1		
13	C-J Pressure	2.8E+10	Pa		
14	Burn on compression fraction	0			
15	Pre-burn bulk modulus	0	Pa		
16	Adiabatic Constant	0			
17	Additional specific energy / unit mass	0	J kg^-1		
18	Begin Time	0	s		
19	End Time	0	s		

Card 1

- EOSID = ID of the keyword, must be unique between the *EOS keywords.
- A = Parameter A.
- B = Parameter B.
- R1 = Parameter R1.
- R2 = Parameter R2.
- OMEG = Parameter W.
- E0 = (C-J Energy/unit mass) × **Density**.
- V0 = Initial Relative Volume, V0.

Card 2a is left blank.

Card 2b is left blank.

Materials keywords

The following are descriptions for *MAT keywords natively supported by projects that use the LS-DYNA system. More generally, any *MAT keyword may be introduced into the project with the help of Commands objects in the Mechanical application (termed **Keyword Snippet** when referring to the LS-DYNA solver). To use it, insert a **Keyword Snippet** under a Geometry body in the Tree Outline. The program will automatically substitute the MID parameter in accordance with the *PART keyword (see below) of the associated body. All other parameters in the **Keyword Snippet** are transcribed literally, overriding any values that would otherwise derive from the Engineering Data workspace.

If the *MAT keyword is entered in a **Keyword Snippet** anywhere else in the Tree Outline, it will be exported literally and Engineering Data EOS information will also be exported, if present. This practice is not recommended, however, and a warning is provided in the header of **Keyword Snippet** objects when detected.

Note:

When writing the optional Viscoelastic cards from the LS-DYNA analysis system a conversion is made to the Prony series terms. The Relative Moduli are converted to Shear\Bulk Moduli by applying the Initial Shear\Bulk Modulus as a scale factor, and the Relaxation Times are converted to Damping Coefficients by inversion.

$G_i=\alpha_i^G G_0$ and $\beta_i^G=1/\tau_i^G$	$K_i=\alpha_i^K K_0, \beta_i^K=1/\tau_i^K$
Where:	Where:
G_i = Shear Moduli	K_i = Bulk Moduli
α_i^G = Relative Moduli	α_i^K = Relative Moduli
G_0 = Initial Shear Modulus	K_0 = Initial Bulk Modulus
τ_i^G = Relaxation Times	τ_i^K = Relaxation Times
β_i^G = Damping Coefficients	β_i^K = Damping Coefficients

The Initial Shear\Bulk Modulus for each material model is determined using the definition in Hyperelasticity in the *Material Reference* or using a specific input field in Engineering Data.

*MAT_3-PARAMETER_BARLAT

This material law is used for the Bilinear 3 Parameter Barlat hardening and exponential 3 parameter Barlat hardening models.

Card 1

- MID = ID of the material. Must be unique between the material keyword definitions.
- RO = density of the material from the Engineering Data workspace.
- E = Young's modulus of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.

- PR = Poisson's Ratio of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.
- HR = Hardening Type, set to 2 for Swift or 5 for Ghosh when the model is exponential, or to 1 when the model is bilinear.
- P1 = Material Parameter, set to Hardening Constant K from Engineering Data when the model is exponential, or to the tangent modulus when the model is bilinear.
- P2 = Material Parameter, set to Hardening Exponent *n* from Engineering Data when the model is exponential, or to the yield stress when the model is bilinear.
- ITER = Iteration flag, set to 0.

Card 2

- M = Barlat exponent from Engineering data.
- R00 = Lankford parameter in 0 degree direction.
- R45 = Lankford parameter in 45 degree direction.
- R90 = Lankford parameter in 90 degree direction.

Card 3

- AOPT = Coordinate System ID of body Coordinate System in Mechanical.
- C = Strain Rate Constant.
- P = Strain Rate Constant.
- EPSO = Reference Strain Rate.

*MAT_BARLAT_ANISOTROPIC_PLASTICITY

Card 1

- MID = ID of the material. Must be unique between the material keyword definitions.
- RO = density of the material from the Engineering Data workspace.
- E = Young's modulus of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.
- PR = Poisson's Ratio of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.
- k = Hardening Constant from Engineering Data
- n = Hardening Exponent from Engineering Data
- m = Barlat Exponent from Engineering Data
- e0 = Initial Yield Strain From Engineering Data

Card 2

- A = Barlat Anisotropic Coefficient A from Engineering data.
- B = Barlat Anisotropic Coefficient B from Engineering data.
- C = Barlat Anisotropic Coefficient C from Engineering data.
- F = Barlat Anisotropic Coefficient F from Engineering data.
- G = Barlat Anisotropic Coefficient G from Engineering data.
- H = Barlat Anisotropic Coefficient H from Engineering data.

Card 3

- AOPT = Coordinate System ID of body Coordinate System in Mechanical.

*MAT_ADD_EROSION

ADD_EROSION is added to a given material when a failure model is defined in Engineering Data, to a material that doesn't support the defined failure model.

Card

- MID = ID of material for which this failure model applies.
- SIGP1 = **Principal Stress Failure** , if present. Otherwise it is 0.
- MXEPS = **Maximum Principal Strain** , if present. Otherwise it is 0.
- EPSSH = **Maximum Shear Strain** , if present. Otherwise it is 0.
- EFFEPS = **Maximum Equivalent Plastic Strain EPS** , if present. Otherwise it is 0.
- MNPRES = **Maximum Tensile Pressure** , if present. Otherwise it is 0.

*MAT_ARRUDA_BOYCE_RUBBER (or *MAT_127)

- MID = ID of the material type. Must be unique between the material keyword definitions.
- RO = Density of the material from the Engineering Data workspace.
- K = Bulk modulus of the material, calculated from incompressibility parameter.
- G = Initial shear modulus of the material from the Engineering Data workspace.

Optional Viscoelastic Constants Cards

- These optional cards will be written if the constants have been defined using the Prony Shear Relaxation properties in Engineering Data

*MAT_BLATZ-KO_RUBBER (or *MAT_007)

Blatz-Ko materials are only for rubber materials under compression.

Poisson's ratio (NUXY) is automatically set to 0.463 by LS-DYNA, so only DENS and GXY are required.

Card1

- MID = ID of the material type. Must be unique between the material keyword definitions.
- RO = Density of the material.
- G = Initial Shear modulus of material.

*MAT_ELASTIC (or *MAT_001)

Specifies isotropic elastic materials. It is available for beam, shell and solid elements. This keyword is used if the selected material includes the Isotropic Elasticity strength model and the **Stiffness Behavior** is set to **Deformable** in the **Definition** section of the body.

Card

- MID = ID of material type. Must be unique between the material keyword definitions.
- RO = density of the material from the Engineering Data workspace.
- E = Young's modulus of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.
- PR = Poisson's ratio of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.

*MAT_ENHANCED_COMPOSITE_DAMAGE (or *MAT_054)

This material keyword is written for orthotropic materials when the shell layered composite damage model in **Analysis Settings** is set to Enhanced Composite Damage Model.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- EA = Young's Modulus X direction from the Orthotropic Elasticity model.
- EB = Young's Modulus Y direction from the Orthotropic Elasticity model.
- EC = Young's Modulus Z direction from the Orthotropic Elasticity model.
- PRBA = Poisson's Ratio XY from the Orthotropic Elasticity model multiplied by Young's Modulus Y / Young's Modulus X.

- PRCA = Poisson's Ratio YZ from the Orthotropic Elasticity model multiplied by Young's Modulus Z / Young's Modulus X.
- PRCB = Poisson's Ratio XZ from the Orthotropic Elasticity model multiplied by Young's Modulus Z / Young's Modulus Y.

Card2

- GAB = Shear Modulus XY from the Orthotropic Elasticity model.
- GBC = Shear Modulus YZ from the Orthotropic Elasticity model.
- GCA = Shear Modulus XZ from the Orthotropic Elasticity model.
- AOPT =
  - 0 (default). When this parameter is set to zero the locally orthotropic material axes are determined from three element nodes. The first node specifies the local origin, the second specifies one of the axes and the third specifies the plane on which the axis rests.
  - - ID of local coordinate system assigned to the body with this material model.

Card 3 is left blank.

Card 4 is left blank.

Card 5 is left blank.

Card 6

- XC = Compressive X of the Orthotropic Stress Limits definition, if present. Otherwise it is 0.
- XT = Tensile X of the Orthotropic Stress Limits definition, if present. Otherwise it is 0.
- YC = Compressive Y of the Orthotropic Stress Limits definition, if present. Otherwise it is 0.
- YT = Tensile Y of the Orthotropic Stress Limits definition, if present. Otherwise it is 0.
- SC = Shear XY of the Orthotropic Stress Limits definition, if present. Otherwise it is 0.

*MAT_HIGH_EXPLOSIVE_BURN

Allows for modeling of the detonation of a high explosive. This keyword is created when an Explosive JWL Equation of State is present in the properties of a material that is used in the simulation. It is used in conjunction with the *EOS_JWL (p. 165) keyword and uses some of the fields defined in the Explosive JWL property.

Card 1

- MID = ID of material type, must be unique between the material keyword definitions.
- R0 = Density.
- D = C-J Detonation Velocity.

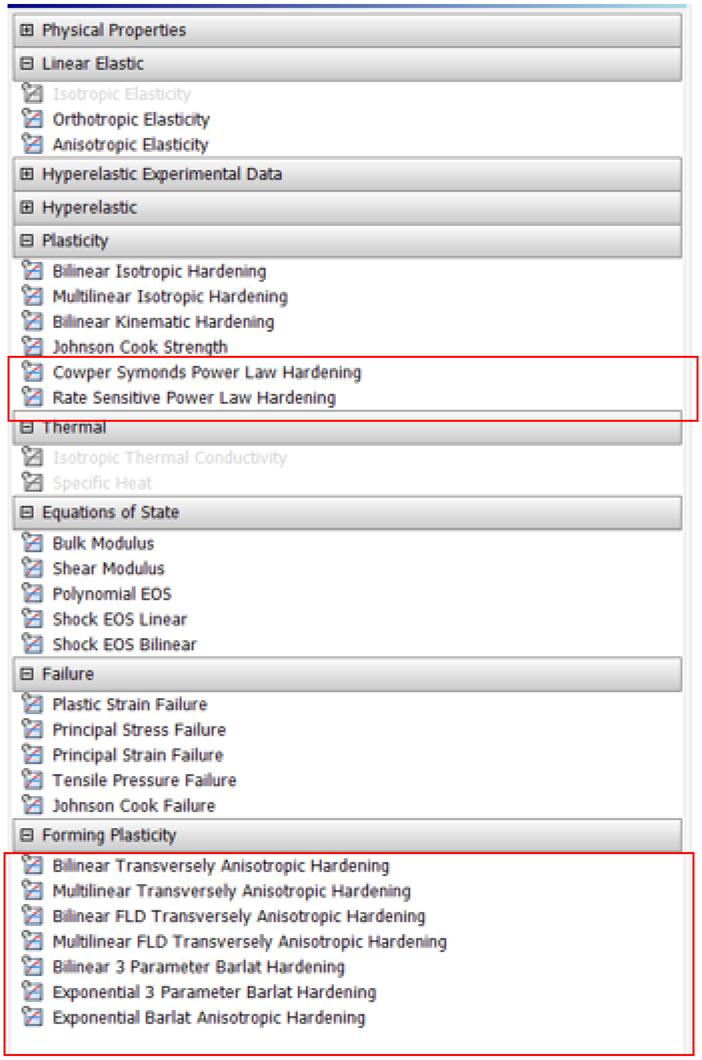
- PCJ = C-J Pressure.
- BETA =
  - 0: if Burn on compression fraction $\neq$ 0.
  - 2: if Burn on compression fraction = 0.
- K = Pre-burn bulk modulus
- G = 0.
- SIGY = 0.

*MAT_LAMINATED_COMPOSITE_FABRIC

This material keyword is written for orthotropic materials when the shell layered composite damage model in **Analysis Settings** is set to Laminated Composite Fabric.

*MAT_FLD_TRANSVERSELY_ANISOTROPIC

This material law is used for the Bilinear transversely anisotropic hardening and multilinear transversely anisotropic hardening models.



Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of the material from the Engineering Data workspace.
- E = Young's modulus of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.
- PR = Poisson's ratio of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.
- SIGY = Yield Strength.
- ETAN = Tangent Modulus.
- R = Anisotropic Hardening Parameter.
- HLCID = 0 when the model is bilinear, or is set to the ID of the curve of effective stress versus plastic strain when the model is multilinear.

- LCFLD is the ID of the curve describing the forming limit diagram.

*MAT_HYPERELASTIC_RUBBER (or *MAT_077_H)

Specifies a general hyperelastic rubber model, optionally combined with viscoelasticity. This keyword is used if the material includes the Mooney-Rivlin, Polynomial or Yeoh hyperelastic strength model and the **Stiffness Behavior** is set to **Deformable** in the **Definition** section of the body.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of the material from the Engineering Data workspace.
- PR = Poisson's ratio of the material from the Engineering Data workspace. Values higher than 0.49 are recommended. Smaller values may not work and should not be used.
- N = 0, specifies that the constants in card 2 will be defined.
- NV = 0. This parameter is not used if N = 0 above.
- G = Shear modulus of the material from the Engineering Data workspace.
- SIGF = 0. This parameter is not used if N = 0 above.

Card2

- C10 = constant C10 from the Engineering Data workspace.
- C01 = constant C01 from the material properties in the Engineering Data. Set to zero for Yeoh models.
- C11 = constant C11 from the Engineering Data workspace. Set to zero for Yeoh models.
- C20 = constant C20 from the Engineering Data workspace.
- C02 = constant C02 from the Engineering Data workspace. Set to zero for Yeoh models.
- C30 = constant C30 from the Engineering Data workspace.

Optional Viscoelastic Constants Cards

- These optional cards will be written if the constants have been defined using the Prony Shear Relaxation properties in Engineering Data

*MAT_JOHNSON_COOK (or *MAT_015)

Defines a Johnson - Cook type of material. Such materials are useful for problems with large variations in strain rates where adiabatic temperature increases due to plastic heating cause material softening. This keyword is used if the material specified includes a Johnson Cook strength model.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.

- RO = density of material.
- G = Shear modulus of material.
- E = Young's modulus of the material (shell elements only).
- PR = Poisson's ratio of the material (shell elements only).

Card2

- A = Initial yield stress from the Johnson Cook strength parameters.
- B = Hardening Constant from the Johnson Cook strength parameters.
- N = Hardening Exponent from the Johnson Cook strength parameters.
- C = Strain Rate Constant from the Johnson Cook strength parameters.
- M = Thermal Softening Exponent from the Johnson Cook strength parameters.
- TM = Melting Temperature from the Johnson Cook strength parameters.
- TR = 15, room temperature.
- EPSO = Reference Strain Rate from the Johnson Cook strength parameters.

Card3

- CP = Specific Heat from the material properties.
- PC = 0 (LS-DYNA default).
- SPALL = 2.0 (LS-DYNA default).
- IT = 0 (LS-DYNA default).
- D1 = D1 parameter of the Johnson Cook failure model definition, if present. Otherwise it is 0.
- D2 = D2 parameter of the Johnson Cook failure model definition, if present. Otherwise it is 0.
- D3 = D3 parameter of the Johnson Cook failure model definition, if present. Otherwise it is 0.
- D4 = D4 parameter of the Johnson Cook failure model definition, if present. Otherwise it is 0.

Card4

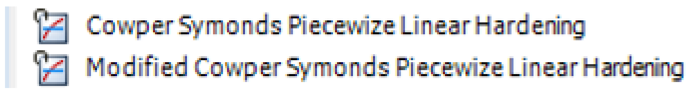
- D5 = D5 parameter of the Johnson Cook failure model definition, if present. Otherwise it is 0.
- C2/P = "Reference Strain Rate (/sec)" parameter of the Johnson Cook failure model definition, if present. Otherwise it is 0.

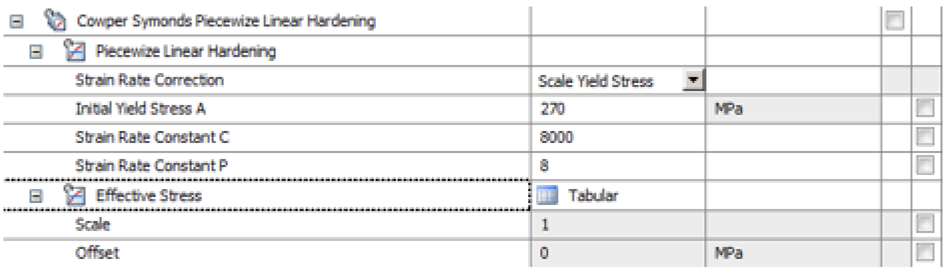
*MAT_MODIFIED_PIECEWISE_LINEAR_PLASTICITY (or *MAT_123)

Enhanced Piecewise Linear model (for shell elements only) that accounts for multiple failure methods:

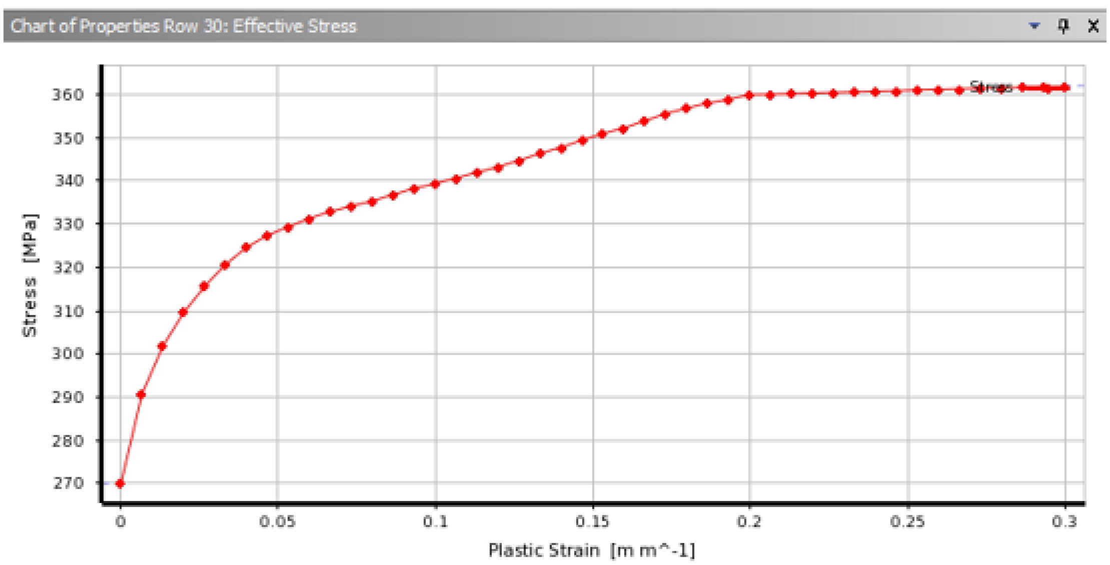
- Effective plastic strain.
- Thinning (through-thickness) plastic strain.
- Major principal in-plane strain.

You can specify the number of through-thickness integration points that must fail before the shell element is deleted. This model is useful in pure bending applications where the center layer may never reach failure strain.





Cowper Symonds Piecewize Linear Hardening		
Piecewize Linear Hardening		
Strain Rate Correction	Scale Yield Stress	
Initial Yield Stress A	270	MPa
Strain Rate Constant C	8000	
Strain Rate Constant P	8	
Effective Stress	Tabular	
Scale	1	
Offset	0	MPa



Card 1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- E = Young's modulus of the material.
- PR = Poisson's ratio of the material.
- SIGY = Yield Strength from the BISO strength model. It is not required for MISO models.

- ETAN = Tangent Modulus from the BISO strength model. It is not required for MISO models.
- FAIL = Maximum Equivalent Plastic Strain EPS parameter of the Plastic Strain failure model, if present. Otherwise it is set to 10E+20.

Card2

- C = Strain Rate Constant C.
- P = Strain Rate Constant P.
- LCSS = ID of the curve defining effective stress versus effective plastic strain.
- LCSR is left blank.
- VP = Formulation for rate effects. Set VP to 0 if the strain rate correction is set to scale yield stress, set it to 1 if strain rate correction is set to viscoplastic.
- EPSTHIN = Thinning Strain at Failure.
- EPSMAJ = Major in Plane Strain At Failure.
- NUMINT is left blank.

Card3 is left blank.

Card4 is left blank.

Card5 is left blank.

*MAT_NULL

Card 1

- MID = ID of material type, must be unique between the material keyword definitions.
- R0 = Density.
- PC = 0.
- MU = 0.
- TEROD = 0.
- CEROD = 0.
- YM = 0.
- PR = 0.

*MAT_OGDEN_RUBBER (or *MAT_077_O)

Specifies the Ogden rubber model, optionally combined with viscoelasticity. This keyword is used if the material includes the Ogden hyperelastic strength model and the **Stiffness Behavior** is set to **Deformable** in the **Definition** section of the body.

For card 1 see *MAT_HYPERELASTIC_RUBBER (p. 174)

Card2

- MU1 = Material Constant MU1 from the Ogden model.
- MU2 = Material Constant MU2 from the Ogden model.
- MU3 = Material Constant MU3 from the Ogden model.
- MU4 = 0.
- MU5 = 0.
- MU6 = 0.
- MU7 = 0.
- MU8 = 0.

Card3

- ALPHA1 = Material Constant A1 from the Ogden model.
- ALPHA2 = Material Constant A2 from the Ogden model.
- ALPHA3 = Material Constant A3 from the Ogden model.
- ALPHA1 = 0.
- ALPHA1 = 0.
- ALPHA1 = 0.
- ALPHA1 = 0.
- ALPHA8 = 0.

Optional Viscoelastic Constants Cards

- These optional cards will be written if the constants have been defined using the Prony Shear Relaxation properties in Engineering Data

*MAT_ORTHOTROPIC_ELASTIC (or *MAT_002)

Specifies the model for an elastic-orthotropic behavior of solids, shells, and thick shells. This keyword is created when the Orthotropic Elasticity property is present in a material that is used. The Poisson's ratios required with this keyword must be in their minor version, however Workbench requires their major versions hence they are converted by multiplying them by the relevant Young's modulus ratios.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- EA = Young's Modulus X direction from the Orthotropic Elasticity model.
- EB = Young's Modulus Y direction from the Orthotropic Elasticity model.
- EC = Young's Modulus Z direction from the Orthotropic Elasticity model.
- PRBA = Poisson's Ratio XY from the Orthotropic Elasticity model multiplied by Young's Modulus Y / Young's Modulus X.
- PRCA = Poisson's Ratio YZ from the Orthotropic Elasticity model multiplied by Young's Modulus Z / Young's Modulus X.
- PRCB = Poisson's Ratio XZ from the Orthotropic Elasticity model multiplied by Young's Modulus Z / Young's Modulus Y.

Card2

- GAB = Shear Modulus XY from the Orthotropic Elasticity model.
- GBC = Shear Modulus YZ from the Orthotropic Elasticity model.
- GCA = Shear Modulus XZ from the Orthotropic Elasticity model.
- AOPT =
  - 0 (default). When this parameter is set to zero the locally orthotropic material axes are determined from three element nodes. The first node specifies the local origin, the second specifies one of the axes and the third specifies the plane on which the axis rests.
  - - ID of local coordinate system assigned to the body with this material model.

Card3 - mandatory, left blank.

Card4 - mandatory, left blank.

*MAT_ANISOTROPIC_ELASTIC (or *MAT_002_ANIS)

Specifies the model for an elastic-anisotropic behavior of solids, shells, and thick shells. This keyword is created when the Anisotropic Elasticity property is present in a material that is used. Due to symmetry only the upper triangular Cij's need to be defined.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.

- RO = density of material.
- C11 = The 1, 1 term in the 6 × 6 anisotropic constitutive matrix.
- C12 = The 1, 2 term in the 6 × 6 anisotropic constitutive matrix.
- C22 =
- C13 =
- C23 =
- C33 =

Card2

- C14 =
- C24 =
- C34 =
- C44 =
- C15 =
- C25 =
- C35 =
- C45 =

Card3

- C55 =
- C16 =
- C26 =
- C36 =
- C46 =
- C56 =
- C66 = The 6, 6 term in the 6 × 6 anisotropic constitutive matrix.
- AOPT =
  - 0 (default). When this parameter is set to zero the locally orthotropic material axes are determined from three element nodes. The first node specifies the local origin, the second specifies one of the axes and the third specifies the plane on which the axis rests.
  - - ID of local coordinate system assigned to the body with this material model.

Card4 - mandatory, left blank.

Card5 - mandatory, left blank.

*MAT_PIECEWISE_LINEAR_PLASTICITY (or *MAT_24)

Defines elasto-plastic materials with arbitrary stress-strain curve and arbitrary strain rate dependency. This keyword is used if the material specified includes a Multilinear Isotropic Hardening (BISO or MISO) strength model.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- E = Young's modulus of the material.
- PR = Poisson's ratio of the material.
- SIGY = Yield Strength from the BISO strength model. It is not required for MISO models.
- ETAN = Tangent Modulus from the BISO strength model. It is not required for MISO models.
- FAIL = **Maximum Equivalent Plastic Strain EPS** parameter of the Plastic Strain failure model, if present. Otherwise it is set to 10E+20.

Card2

- C = Strain Rate Constant C when used in the Cowper Symonds Piecewise Linear Hardening Engineering Data Material, 0 otherwise.
- P = Strain Rate Constant P when used in the Cowper Symonds Piecewise Linear Hardening Engineering Data Material, 0 otherwise.
- LCSS = ID of the curve that defining effective stress versus effective plastic strain.

Card3 - mandatory, left blank.

Card4 - mandatory, left blank.

*MAT_PLASTIC_KINEMATIC (or *MAT_003)

Specifies isotropic and kinematic hardening plastic behavior in materials. This keyword is created when the Bilinear Kinematic Hardening (BKIN) strength model is present in a material.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- E = Young's modulus of the material.

- PR = Poisson's ratio of the material.
- SIGY = Yield Strength from the BKIN strength model.
- ETAN = Tangent Modulus from the BKIN strength model.
- BETA = 0.

Card2

- SRC = left blank.
- SRP = left blank.
- FS = **Maximum Equivalent Plastic Strain EPS** parameter of the Plastic Strain failure model, if present. Otherwise it is left blank.

*MAT_POWER_LAW_PLASTICITY (or *MAT_018)

Defines an isotropic plasticity model with rate affects modeled by a power hardening law. Power law hardening defined with strength coefficient k and hardening coefficient n .

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- E = Young's modulus of the material.
- PR = Poisson's ratio of the material.
- K = Strength coefficient.
- N = Hardening exponent.
- SRC = Strain rate parameter C. If zero, rate effects are ignored.
- SRP = Strain rate parameter, P. If zero, rate effects are ignored.

Card 2

- SIGY = Initial yield stress.
- EPSF = Maximum Equivalent Plastic Strain EPS parameter of the Plastic Strain failure model, if present. Otherwise it is left blank.
- VP = Formulation for rate effects. Set VP to 0 if the strain rate correction is set to scale yield stress. Set it to 1 if the strain rate correction is set to viscoplastic.

*MAT_RATE_SENSITIVE_POWERLAW_PLASTICITY (or *MAT_064)

This specialized model is used specifically for superplastic forming.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- E = Young's modulus of the material.
- PR = Poisson's ratio of the material.
- K = Hardening Constant.
- M = Hardening exponent.
- N = Strain Rate Constant.
- EO = Reference Strain Rate
- VP = Formulation for rate effects. Set VP to 0 if the strain rate correction is set to scale yield stress. Set it to 1 if strain rate correction is set to viscoplastic
- EPSO is left blank

*MAT_RIGID (or *MAT_020)

Specifies materials for rigid bodies. This keyword is created when the **Stiffness Behavior** is set to **Rigid** under the **Definition** section of the body. Any strength or EOS material properties defined are ignored.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- E = Young's modulus of the material.
- PR = Poisson's ratio of the material.

Card2

- CMO =
  - 0 if there are no constraints on the rigid body.
  - -1 if rigid body is constrained in any way.
- CON1 =
  - 0 if there are no constraints on the rigid body.
  - = Local Coordinate System ID if associated with the constraint. Otherwise it is set to 0.
- CON2 =
  - 0 if there are no constraints on the rigid body.

  - = 111111 if the body is constrained with a fixed support or with a combination of a simple support and a fixed rotation.
  - = 111000 if the body is constrained with a simple support.
  - = 000111 if the body is constrained with a fixed rotation.

Card3

- LCO = CON1 if non-zero. Otherwise it will remain blank.

*MAT_SIMPLIFIED_JOHNSON_COOK (or *MAT_098)

Defines a Johnson - Cook type of material. Such materials are useful for problems with large variations in strain rates where adiabatic temperature increases due to plastic heating cause material softening. This keyword is used if the material specified includes a Johnson Cook strength model without an Equation Of State.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of material.
- E = Young's modulus of the material.
- PR = Poisson's ratio of the material.

Card2

- A= Initial yield stress from the Johnson Cook strength parameters.
- B = Hardening Constant from the Johnson Cook strength parameters.
- N = Hardening Exponent from the Johnson Cook strength parameters.
- C = Strain Rate Constant from the Johnson Cook strength parameters.
- PSFAIL = **Maximum Equivalent Plastic Strain EPS** parameter of the Plastic Strain failure model, if present. Otherwise it is set to 10E+20.
- SIGMAX = 0. Not used.
- SIGSAT = 0. Not used.
- EPSO = Reference Strain Rate from the Johnson Cook strength parameters.

*MAT_SPRING_INELASTIC

This material model is used for springs with a tension only or compression only behavior .

Card

- MID = ID of material type. Must be unique between the material keyword definitions.

- LCFD = ID for curve describing force versus displacement.
- KU is unused.
- CTF = flag for compression/tension behavior. It is set to -1.0 for tension only and to 1.0 for compression only.

*MAT_SPRING_NONLINEAR_ELASTIC

This material model is used for springs when the force versus displacement curve is tabular and the behavior is set to both.

Card

- MID = ID of material type. Must be unique between the material keyword definitions.
- LCD = ID for curve describing force versus displacement.

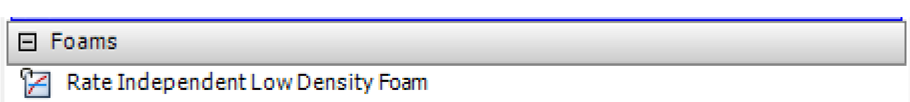
*MAT_TRANSVERSELY_ANISOTROPIC_ELASTIC_PLASTIC

This material law is used for the Bilinear transversely anisotropic hardening and multilinear transversely anisotropic hardening models.

Card1

- MID = ID of material type, must be unique between the material keyword definitions.
- RO = density of the material from the Engineering Data workspace.
- E = Young's modulus of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.
- PR = Poisson's ratio of the material from the Engineering Data workspace, either specified directly or calculated from Bulk and Shear moduli.
- SIGY = Yield Strength.
- ETAN = Tangent Modulus.
- R = Anisotropic Hardening Parameter.
- HLCID = 0 when the model is bilinear, or is set to the ID of the curve of effective stress versus plastic strain when the model is multilinear.

*MAT_LOW_DENSITY_FOAM



5.8. Mesh Definition

*NODE

Defines nodes. All the parameters are obtained from mesh definitions of the model.

Card

- NID = ID of the node.
- X = x coordinate.
- Y = y coordinate.
- Z = z coordinate.

*ELEMENT_BEAM

Specifies beam elements.

Card

- EID = ID of the element.
- PID = ID of the part it belongs to.
- N1 = ID of nodal point 1.
- N2 = ID of nodal point 2.
- N3 = ID of nodal point 3, used for cross section orientation.

*ELEMENT_SHELL

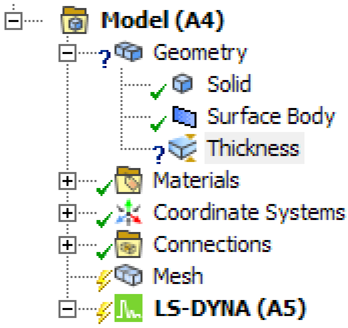
Specifies three, four, six and eight noded shell elements.

Card

- EID = ID of the element.
- PID = ID of the part it belongs to.
- N1 = ID of nodal point 1.
- N2 = ID of nodal point 2.
- N3 = ID of nodal point 3.
- N4 = ID of nodal point 4.
- N5-8 = ID of mid side nodes for six and eight noded shells.

*ELEMENT_SHELL_THICKNESS_OFFSET

Surface body thicknesses properties can be defined on faces of surface bodies using the **Thickness** object in the **Geometry** . This keyword defines scoped surface body thickness.



Card1 - the same as *ELEMENT_SHELL

Card2

- THIC1 = **Thickness** field of the **Thickness** object.
- THIC2 = **Thickness** field of the **Thickness** object.
- THIC3 = **Thickness** field of the **Thickness** object.
- THIC4 = **Thickness** field of the **Thickness** object.

Card3

- OFFSET = value calculated from the **Offset Type** field

  of the **Thickness** object. if **Offset Type** =

  – Middle, it equals zero.

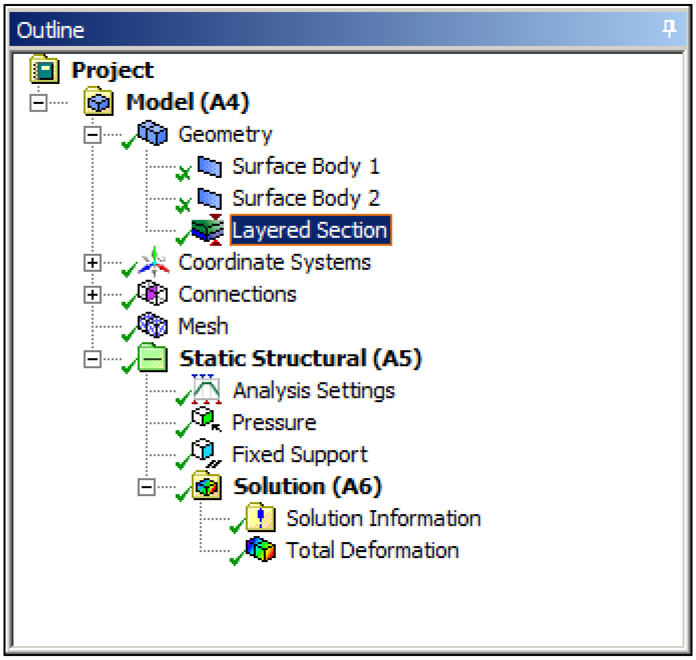
  – Top, it is equal to half of the thickness as a negative number.

  – Bottom, it is equal to half of the thickness.

  – User defined, it is equal to the value defined in the field **Membrane offset** .

*ELEMENT_SHELL_COMPOSITE_OFFSET

The Keyword *ELEMENT_SHELL_COMPOSITE_OFFSET allows the definition of material properties, thickness and orientation angle per thickness IP on an element basis and is used in support of ACP ply based modeling. This keyword is written in the context of layered sections. Layered section properties can be defined on faces of surface bodies using the **Layered Section** object in the **Geometry** .



The first card is identical to *ELEMENT_SHELL.

Card2

- OFFSET: value calculated from the **Offset Type** field.

If offset type is set to Middle, it equals to zero.

If offset type is set to Top, it equals to half of the thickness as a negative number.

If offset type is set to Bottom, it equals to half of the thickness.

If offset type is set to User defined, it is equal to the value defined in the field Membrane offset.

Card3

Defines the property of two layers. Card3 is repeated as many times as required to specify all the layers in the section. The sequence is starting with the bottommost layer.

- MID1: ID of material in Layer1. Must be unique between the material keyword definitions.
- THICK1: Thickness of Layer1.
- B1: Angle defined in the worksheet for Layer1 projected onto the element surface.
- MID2: ID of material in Layer 2. Must be unique between the material keyword definitions.
- THICK2: Thickness of Layer2.
- B2: Angle defined in the worksheet for Layer2 projected onto the element surface.

*ELEMENT_SOLID

Specifies 3D solid elements including 10-noded tetrahedrons (second order). Apart from the second order case the two cards are combined into one.

Card1

- EID = ID of the element.
- PID = ID of the part it belongs to.

Card2

- N1 = ID of nodal point 1.
- N2 = ID of nodal point 2.
- N3 = ID of nodal point 3.
- N4 = ID of nodal point 4.
- .
- .
- .
- N10 = ID of nodal point 10.

*ELEMENT_SPH

Define a lumped mass element assigned to a nodal point (if the value is negative then the value is assigned to the particle volume).

Card

- NID: Node ID and Element ID are the same for the SPH option.
- PID: Part ID to which this node (element) belongs.
- MASS (or VOLUME):
  - If positive: Mass value.
  - If negative: Volume. The density (rho) will be retrieved from the material card defined in PID. SPH element mass is calculated by abs(MASS) $\times \rho$.

5.9. Coordinate Systems

*DEFINE_COORDINATE_SYSTEM

Specifies a local coordinate system with three points: one at the local origin, one on the local x-axis and one on the local x-y plane.

Card1

- CID = ID of the coordinate system, must be unique.
- XO = global X-coordinate of the origin.
- YO = global Y-coordinate of the origin.
- ZO = global Z-coordinate of the origin.
- XL = global X-coordinate of a point on the local x-axis.
- YL = global Y-coordinate of a point on the local x-axis.
- ZL = global Z-coordinate of a point on the local x-axis.

Card2

- XP = global X-coordinate of a point on the local x-y plane.
- YP = global Y-coordinate of a point on the local x-y plane.
- ZP = global Z-coordinate of a point on the local x-y plane.

*DEFINE_VECTOR

Specifies a vector by defining the coordinates of two points. This keyword defines the local coordinate system with respect to which a *BOUNDARY_PRESCRIBED_MOTION is prescribed. The ID of this coordinate system is specified with parameter CID.

Card

- VID = ID of the vector.
- XT = 0, the local x-coordinate of the origin of the coordinate system specified with CID below.
- YT = 0, the local y-coordinate of the origin of the coordinate system specified with CID below.
- ZT = 0, the local z-coordinate of the origin of the coordinate system specified with CID below.
- XH = 1 if the vector has a component in the x direction of the coordinate system specified with CID. Otherwise, this is set to 0.
- YH = 1 if the vector has a component in the y direction of the coordinate system specified with CID. Otherwise, this is set to 0.

- ZH = 1 if the vector has a component in the z direction of the coordinate system specified with CID. Otherwise, this is set to 0.
- CID = ID of the coordinate system used to define the vector. If no coordinate system is specified this parameter is set to 0 to specify the global coordinate system.

5.10. Components and Named Selections

*SET_NODE_LIST

Defines a set of nodes. Card2 is repeated as many times as required to specify all the node IDs in the set.

Card1

- SID = ID of the set.

Card2

- NID1-NID8 = IDs for eight of the nodes in the set.

*SET_PART_LIST

Defines a set of parts. Card2 is repeated as many times as required to specify all the part IDs in the set.

Card1

- SID = ID of the set.

Card2

- PID1-PID8 = IDs for eight of the parts in the set.

*SET_SEGMENT

Defines triangular and quadrilateral segments. Card2 is repeated as many times as required to specify all the segments in the set.

Card1

- SID = ID of the set.

Card2

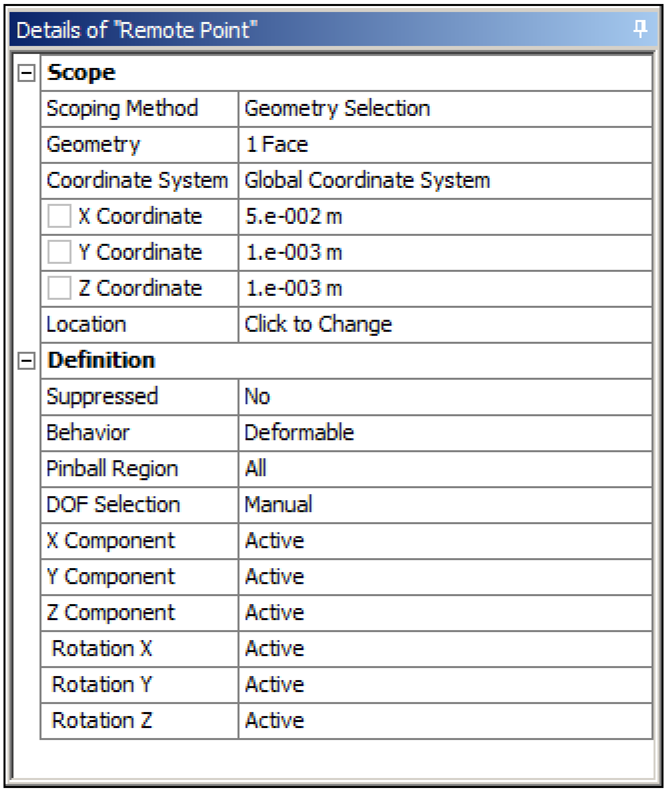
- N1-N4 = IDs of nodes that define one of the segments. For triangular segments N4=N3.

5.11. Remote Points and Point Masses

*CONSTRAINED_NODAL_RIGID_BODY

This keyword is generated for remote points. Remote points are a way of abstracting a connection to a solid model, be it a vertex, edge, face, body, or node, to a point in space (specified by Location).

Remote Points are akin to the various remote loads available in the Mechanical application. Remote boundary conditions create remote points in space behind the scenes, or internally, whereas the Remote Point objects define a specific point in space only. A Remote Point is converted to a rigid constraint (nodal rigid body) independent of the stiffness behavior.



the location set in the Remote Point **Scope** is not used in the input file definition.

Card1

- PID = ID of the Rigid Body. It is set in the LS-DYNA solver and does not reflect the ID specified in the remote point definition.
- NSID identifies a set of nodes that are to be defined as a rigid body. This set of nodes is based on the scoped entities. The set consists of nodes from several different deformable parts.
- PNODE = 0. This is not used in the exported file.
- DRFLAG = The value is calculated from the translational active degrees of freedom when the DOF Selection in the Remote Point definition is set to Manual. It allows you to deactivate certain degrees of freedom in the rigid body definition. DRFLAG =
  - 1, when X Component is inactive.

- 2, when Y Component is inactive.
- 3, When Z Component is inactive.
- 4, when X and Y Component are inactive.
- 5, when Y and Z Component are inactive.
- 6, when Z and X Component are inactive.
- 7, when X, Y, and Z Components are inactive.

- RRFLAG = The value calculated from the rotational active degrees of freedom when the **DOF Selection** in the Remote Point Definition is set to Manual. It allows you to deactivate certain degrees of freedom in the rigid body definition. RRFLAG =
  - 1, when Rotation X is inactive.
  - 2, when Rotation Y is inactive.
  - 3, when Rotation Z is inactive.
  - 4, when Rotation X and Y are inactive.
  - 5, when Rotation Y and Z are inactive.
  - 6, when Rotation Z and X are inactive.
  - 7, when Rotation X, Y, and Z are inactive.

*CONSTRAINED_NODAL_RIGID_BODY_INERTIA

This keyword is generated for point masses. Point masses use a remote point for their definition, or can be applied on a remote point. See *CONSTRAINED_NODAL_RIGID_BODY for additional information.

Card1

- PID = ID of the Rigid Body. It is set in the LS-DYNA solver and does not reflect the ID specified in the remote point/Point Mass definition.
- NSID identifies a set of nodes that are to be defined as a rigid body. This set of nodes is based on the scoped entities. The set consists of nodes from several different deformable parts.
- PNODE = 0. This is not used in the exported file.
- DRFLAG = 0. All Translational degrees of freedom are active in the rigid body definition.
- RRFLAG = 0. All Rotational degrees of freedom are active in the rigid body definition.

Card2

- XC = **X Coordinate** from the **Scope** of the Point Mass object.
- YC = **Y Coordinate** from the **Scope** of the Point Mass object.

- ZC = **Z Coordinate** from the **Scope** of the Point Mass object.
- TM = **Mass** from the **Scope** of the Point Mass object.

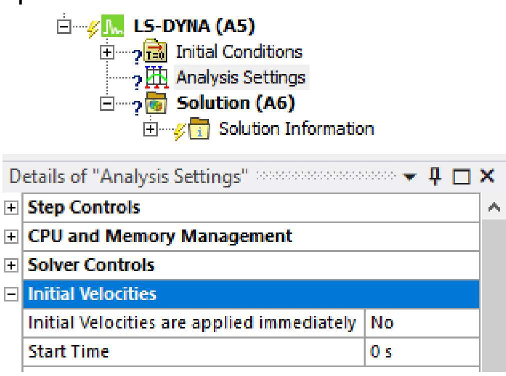
Card3

- IXX = **Mass Moment Of Inertia X** from the **Definition** of the Point Mass object.
- IXY = 0
- IXZ = 0
- IYY = **Mass Moment Of Inertia Y** from the **Definition** of the Point Mass object.
- IYZ = 0
- IZZ = **Mass Moment Of Inertia Z** from the **Definition** of the Point Mass object.

5.12. Initial Conditions

*INITIAL_VELOCITY_GENERATION

Specifies initial translational and rotational velocities.



You can apply initial velocities immediately or delay the initial velocity. To delay the initial velocity:

1. Set **Initial Velocities are applied immediately** to No in the Initial Velocities section of the Analysis Settings object.
2. Enter a **Start Time**. In this case, the LS-DYNA system will write the card *INITIAL_VELOCITY_GENERATION_START_TIME to the input file.

When applying a delayed Initial Velocity to an SPH part, the LS-DYNA system will write two *INITIAL_VELOCITY_GENERATION cards, one with PHASE=0 and one with PHASE=1.

Card1

- ID = ID of part where the initial velocity is applied.
- STYP = 2, the velocity is applied to a whole part. In Workbench initial velocities can only be applied to whole parts.

- OMEGA = angular velocity about the rotational axis.
- VX = initial translational velocity in the x direction.
- VY = initial translational velocity in the y direction.
- VZ = initial translational velocity in the z direction.
- IVATN = 0 (default) contact bodies of a multibody part are not assigned the initial velocities of the target part.
- ICID = Local coordinate system ID. The specified velocities are in the local system.

Card2

- XC = 0. x coordinate of the origin of the applied coordinate system.
- YC = 0. y coordinate of the origin of the applied coordinate system.
- ZC = 0. z coordinate of the origin of the applied coordinate system.
- NX = x-direction cosine.
- NY = y-direction cosine.
- NZ = z-direction cosine.
- PHASE = 0 (default), velocities are applied immediately.
- IRIGID = 0: Option to overwrite or automatically set rigid body velocities defined on the *PART_INERTIA and *CONSTRAINED_NODAL_RIGID_BODY _INERTIA cards.

*CHANGE_VELOCITY_GENERATION

Modifies initial translational and rotational velocities in Small and Full Restarts.

Card1

- ID = ID of part where the initial velocity is applied.
- STYP = 2, the velocity is applied to a whole part. In Workbench initial velocities can only be applied to whole parts.
- OMEGA = angular velocity about the rotational axis.
- VX = initial translational velocity in the x direction.
- VY = initial translational velocity in the y direction.
- VZ = initial translational velocity in the z direction.
- IVATN = 0 (default) contact bodies of a multibody part are not assigned the initial velocities of the target part.
- ICID = Local coordinate system ID. The specified velocities are in the local system.

Card2

- XC = 0. x coordinate of the origin of the applied coordinate system.
- YC = 0. y coordinate of the origin of the applied coordinate system.
- ZC = 0. z coordinate of the origin of the applied coordinate system.
- NX = x-direction cosine.
- NY = y-direction cosine.
- NZ = z-direction cosine.
- PHASE = 0 (default), velocities are applied immediately.
- IRIGID = 0: Option to overwrite or automatically set rigid body velocities defined on the *PART_INERTIA and *CONSTRAINED_NODAL_RIGID_BODY _INERTIA cards.

*INITIAL_VELOCITY_RIGID_BODY

Specifies initial translational and rotational velocities at the center of gravity for rigid bodies.

Card

- PID = ID of the rigid body.
- VX = initial translational velocity in the x direction.
- VY = initial translational velocity in the y direction.
- VZ = initial translational velocity in the z direction.
- VXR = initial rotational velocity around the x-axis.
- VYR = initial rotational velocity around the y-axis.
- VZR = initial rotational velocity around the z-axis.

5.13. Contacts and Body Interactions

*CONTACT_AUTOMATIC_GENERAL

Specifies friction or frictionless contacts between line bodies (beams). This keyword is created if the contact is specified using **Body Interactions** and the geometry contains line bodies.

All the parameter cards are the same as in *CONTACT_AUTOMATIC_SINGLE_SURFACE.

*CONTACT_AUTOMATIC_NODES_TO_SURFACE

Specifies nodes-to-surface friction or frictionless contacts. This keyword is created if the contact is specified using a **Contact Region** and the **Behavior** is set to **Asymmetric** .

Card1 - mandatory

- SSID = ID for the set of contact nodes involved in the contact.
- MSID = ID for the set of target segments involved in the contact.
- SSTYP = 4, the contact entities for the contact are nodes or SPH particles.
- MSTYP = 0, the target entities for the contact are segments.
- SBOXID, MBOXID, SPR and MPR are the same as in *CONTACT_AUTOMATIC_SINGLE_SURFACE.

Parameter Card2 and Card3 is the same as in *CONTACT_AUTOMATIC_SINGLE_SURFACE.

*CONTACT_AUTOMATIC_SINGLE_SURFACE

Specifies friction or frictionless contacts between parts. This keyword is created if the contact is specified using **Body Interactions** .

Card1 - mandatory

- SSID = ID for the set of parts created for the bodies in the **Body Interaction** . If the contact is applied to all the bodies in the geometry then this parameter is set to 0.
- MSID = 0.
- SSTYP =2, the contact entities are parts. If the contact is applied to all the bodies in the geometry then this parameter is set to 5.
- MSTYP = 2, the target entities are parts. If the contact is applied to all the bodies in the geometry then this parameter is set to 0.
- SBOXID = It is not used, will be left blank.
- MBOXID = It is not used, will be left blank.
- SPR = 1 (constant) requests that forces on the contact side of the contact be included in the results files NCFORC (ASCII) and INTFOR (binary).
- MPR = 1 (constant) requests that forces on the target side of the contact be included in the results files NCFORC (ASCII) and INTFOR (binary). T

Card2 - mandatory

- FS = Friction Coefficient value from the inputs for frictional contact.
- FD = Dynamic Coefficient value from the inputs for frictional contact.
- DC = Decay Constant value from the inputs for frictional contact.
- VC = 0 (LS-DYNA default).
- VDC = 10 (constant). This parameter specifies the percentage of the critical viscous damping coefficient to be used in order to avoid undesirable oscillation in the contact.

Card3 - mandatory, left blank for defaults to be used.

Card A is the same as for *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE.

*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE

Defines specific surface-to-surface friction or frictionless contacts. This keyword is created if the contact is specified using a **Contact Region** and the **Behavior** is set to **Symmetric** .

Card1 - mandatory

- SSID = ID for the set of contact segments involved in the contact.
- MSID = ID for the set of target segments involved in the contact.
- SSTYP = 0, the contact entities for the contact are segments.
- MSTYP = 0, the target entities for the contact are segments.
- SBOXID, MBOXID, SPR and MPR are the same as in *CONTACT_AUTOMATIC_SINGLE_SURFACE.

Parameter Card2 and Card3 are the same as in *CONTACT_AUTOMATIC_SINGLE_SURFACE.

Card A

- SOFT = 2 except for asymmetric contacts like NODES_TO_SURFACE and unbreakable bonded contacts for which it is set to 1.
- SOFSCL = left blank, the default value of 0.1 will be used. This scale factor is used to determine the stiffness of the interface when SOFT is set to 1. For SOFT = 2 scale factor SLSFAC (see *CONTROL_CONTACT) is used instead.
- LCIDAB = left blank.
- MAXPAR= left blank.
- SBOPT = 3.
- DEPTH = 5.

*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK

Specifies breakable symmetric bonded contacts. This keyword is created for **Bonded** contact when the **Breakable** option is set to **Stress Criteria** and the contact **Behavior** is set to **Symmetric** .

Card 1 is the same as in *CONTACT_TIED_SURFACE_TO_SURFACE_OFFSET.

Card2 - mandatory

- FS = Normal Stress Limit value for the bonded contact.
- FD = Shear Stress Limit value for the bonded contact.
- DC = 0 (LS-DYNA default). This parameter is not required for bonded contacts.

- VC and VDC are the same as in *CONTACT_AUTOMATIC_SINGLE_SURFACE.

Card3 - mandatory, is left blank.

Card A is the same as for *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE.

*CONTACT_ONEWAY_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK

Specifies breakable asymmetric bonded contacts. This keyword is created for **Bonded** contact when the **Breakable** option is set to **Stress Criteria** and the contact **Behavior** is set to **Asymmetric** .

Parameter cards are the same as in *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK.

Card A is not used for this keyword.

*CONTACT_TIED_NODES_TO_SURFACE_OFFSET

Specifies non-breakable asymmetric bonded contacts. This keyword is created for **Bonded** contacts that are not designated as **Breakable** whose **Behavior** is set to **Asymmetric** . This keyword is not used for Body Interactions as these types of contacts are always symmetric.

Card1 - mandatory

- SSID = ID for the set of contact nodes involved in the contact.
- MSID = ID for the set of target segment or for the set of parts involved in the contact.
- SSTYP = 4. SSID indicates the ID for a set of nodes.
- MSTYP = 0, MSID indicates the ID for a set of segments.
- SBOXID, MBOXID, SPR and MPR are the same as in *CONTACT_AUTOMATIC_SINGLE_SURFACE.

Card 2 left blank.

Card 3

- SFS = left blank, the default value of 1.0 will be used. Default contact penalty stiffness scale factor for SLSFAC (see *CONTROL_CONTACT).
- SFM= left blank, the default value of 1.0 will be used. Default target penalty stiffness scale factor for SLSFAC (see *CONTROL_CONTACT).
- SST = the negative value of:

  $$\frac{\text{Maximum Offset}}{0.6}$$

  "Maximum Offset" is the Definition parameter available for bonded contacts and body interactions. "Maximum Offset" is obtained from the inputs of the **Contact Region** of **Bonded** type.
- MST = SST.

*CONTACT_TIED_NODES_TO_SURFACE_CONSTRAINED_OFFSET

Specifies non-breakable asymmetric bonded contacts. This keyword is created for Bonded contacts that are not designated as Breakable whose Behavior is set to Asymmetric and when the contact Formulation is set to MPC. This keyword is not used for Body Interactions as these types of contacts are always symmetric.

The card is identical to CONTACT_TIED_NODES_TO_SURFACE_ OFFSET.

*CONTACT_TIED_SURFACE_TO_SURFACE_OFFSET

Specifies general non-breakable bonded contacts that are symmetric. This keyword is created for **Bonded** and non-breakable contacts which are defined by **Contact Regions** that are **Bonded** , non-breakable and whose **Behavior** is set to **Symmetric** .

Card1 - mandatory

- SSID = ID for a set of contact segments or a set of parts involved in the contact.
- MSID = ID for the set of target segments or the set of parts involved in the contact.
- SSTYP = specifies whether the ID used in SSID represents parts or segments. It is set to 0 if SSID represents a set of segments and 2 if it represents a set of parts.
- MSTYP = SSTYP.
- SBOXID, MBOXID, SPR and MPR are the same as in *CONTACT_AUTOMATIC_SINGLE_SURFACE.

Cards 2 and 3 are the same as in *CONTACT_TIED_NODES_TO_SURFACE_OFFSET.

Card A is the same as for *CONTACT_AUTOMATIC_SURFACE_TO_SURFACE.

*CONTACT_TIED_SURFACE_TO_SURFACE_CONSTRAINED_OFFSET

Specifies general non-breakable bonded contacts that are symmetric. This keyword is created for Bonded and non-breakable contacts which are defined by Contact Regions that are Bonded, non-breakable and whose Behavior is set to Symmetric and when the contact Formulation is set to MPC.

The card is identical to CONTACT_TIED_SURFACE_TO_SURFACE_ OFFSET.

*CONSTRAINED_SPOTWELD

Specifies spot welds between non-contiguous nodal pairs of shell elements. This keyword is created when a spot weld contact is defined in the Mechanical application.

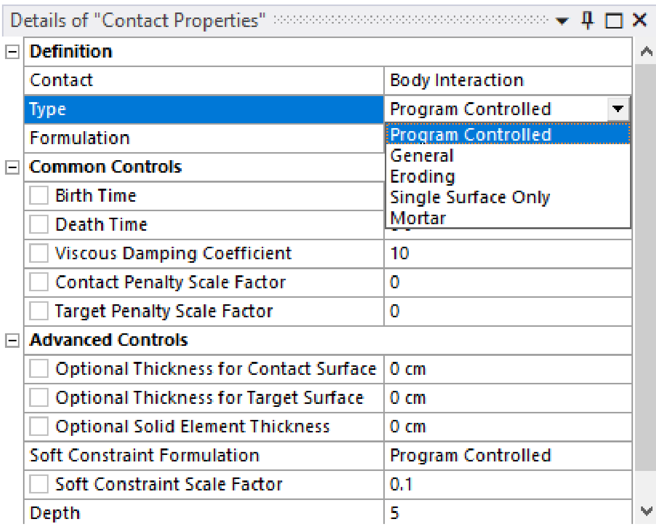
Card

- N1 = ID of the first node used in the weld.
- N2 = ID of the second node present in the weld.
- SN = Normal force at weld failure.

- SS = Shear force at weld failure.
- N = Exponent of normal force.
- M = Exponent of shear force.

5.13.1. Keywords Created from the Contact Properties Object

The **Contact Properties** object found on the LSDYNA Pre tab allows you to modify the default generated values (the type of the contact) and specify additional values.



*CONTACT_ERODING_SINGLE_SURFACE is written if the contact is specified using a body interaction.

The following keywords are written if the contact is specified using a contact region, and the indicated conditions exist.

*CONTACT_ERODING_NODES_TO_SURFACE is written if the Contact Properties **Type** field is set to Eroding and the contact is asymmetric.

*CONTACT_ERODING_SURFACE_TO_SURFACE is written if the Contact Properties **Type** field is set to Eroding and the contact is symmetric.

*CONTACT_FORMING_SURFACE_TO_SURFACE is written if the Contact Properties **Type** field is set to Forming and the contact is symmetric.

*CONTACT_FORMING_ONE_WAY_SURFACE_TO_SURFACE is written if the Contact Properties **Type** field is set to Forming and the contact is asymmetric.

*CONTACT_FORMING_NODES_TO_SURFACE is written if the Contact Properties **Type** field is set to Forming and the contact is asymmetric, and the scoped entities on the contact side are edges.

*CONTACT_INTERFERENCE_SURFACE_TO_SURFACE is written if the Contact Properties **Type** field is set to Interference and the contact is symmetric.

*CONTACT_INTERFERENCE_ONE_WAY_SURFACE_TO_SURFACE is written if the Contact Properties **Type** field is set to Interference and the contact is asymmetric.

*CONTACT_INTERFERENCE_NODES_TO_SURFACE is written if the Contact Properties **Type** field is set to Interference and the contact is asymmetric, and the scoped entities on the contact side are edges.

*CONTACT_TIED_SHELL_EDGE_TO_SURFACE is written if the Contact Properties **Type** field is set to Tied Shell Edge, the contact is asymmetric, and the contact formulation is set to MPC.

*CONTACT_TIED_SHELL_EDGE_TO_SURFACE_BEAM_OFFSET is written if the Contact Properties **Type** field is set to Tied Shell Edge, the contact is asymmetric. This is the default behavior.

*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR_TIED is written if the Contact Properties **Type** field is set to Mortar and the contact is symmetric, bonded, and non breakable.

*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_TIEBREAK_MORTAR is written if the Contact Properties **Type** field is set to Mortar and the contact is symmetric and breakable.

*CONTACT_AUTOMATIC_SURFACE_TO_SURFACE_MORTAR is written if the Contact Properties **Type** field is set to Mortar and the contact is symmetric.

*CONTACT_AUTOMATIC_SINGLE_SURFACE_MORTAR is written if the contact is specified using a Body Interaction and the Contact Properties **Type** field is set to Mortar.

The cards for these contact keywords are as follows:

Card1

- BT = Birth Time from the **Common Controls** section of the **Contact Properties** object.
- DT = Death Time from the **Common Controls** section of the **Contact Properties** object.
- SFS = Contact Penalty Scale Factor from the **Common Controls** section of the **Contact Properties** object.
- SFM = Target Penalty Scale Factor from the **Common Controls** section of the **Contact Properties** object.
- SST = Optional Thickness for Contact Surface from the **Advanced Controls** section of the **Contact Properties** object.
- MST = Optional Thickness for Target Surface from the **Advanced Controls** section of the **Contact Properties** object.
- DEPTH = Depth from the **Advanced Controls** section of the **Contact Properties** object.

Card A is also modified

- SOFT = Soft Constraint Formulation from the **Advanced Controls** section of the **Contact Properties** object.
- SOFTSCL = Soft Constraint Scale Factor from the **Advanced Controls** section of the **Contact Properties** object.

If the contact type is set to Eroding, additional parameters are available to support this formulation.

- ISYM = Symmetry Plane Option from the **Erosion Controls** section of the **Contact Properties** object.

- IADJ = Erosion Node Option from the **Erosion Controls** section of the **Contact Properties** object.
- EROSOP = Solid Elements Treatment from the **Erosion Controls** section of the **Contact Properties** object.

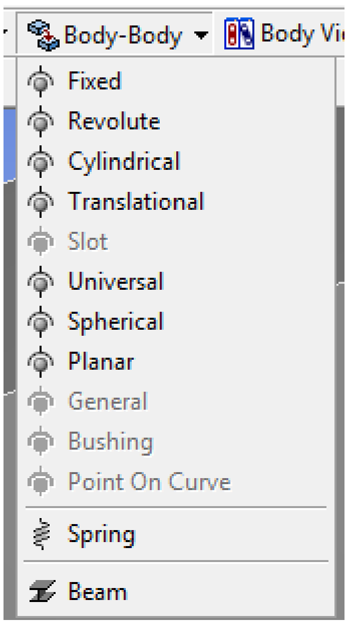
*CONTACT_FORCE_TRANSDUCER_PENALTY

When single surface contacts are used, one or more force transducers are added via the `*CONTACT_FORCE_TRANSDUCER_PENALTY` command. A force transducer does not produce any contact forces and allows you to monitor the contact forces on a subset of parts of the models.

A force transducer is added for each body involved in a body interaction (Frictionless or Frictional).

*CONSTRAINED_LAGRANGE_IN_SOLID

5.14. Kinematic Joints



*CONSTRAINED_JOINT_LOCKING

This keyword is created for a body to body fixed joint in the mechanical GUI.

*CONSTRAINED_JOINT_REVOLUTE

This keyword is created for a body to body revolute joint in the mechanical GUI.

*CONSTRAINED_JOINT_TRANSLATIONAL

This keyword is created for a body to body translational joint in the mechanical GUI.

*CONSTRAINED_JOINT_CYLINDRICAL

This keyword is created for a body to body cylindrical joint in the mechanical GUI.

*CONSTRAINED_JOINT_UNIVERSAL

This keyword is created for a body to body universal joint in the mechanical GUI.

*CONSTRAINED_JOINT_SPHERICAL

This keyword is created for a body to body spherical joint in the mechanical GUI.

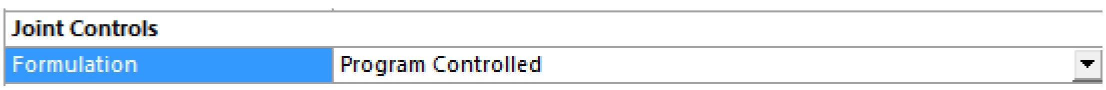
*CONSTRAINED_JOINT_PLANAR

This keyword is created for a body to body planar joint in the mechanical GUI.

*CONTROL_RIGID

Specify the explicit rigid body joint treatment. It can use an implicit formulation which uses Lagrange multipliers or penalty.

- LMF is written according to the selection in the GUI.



5.15. Magnitude and Tabular Data

*DEFINE_CURVE

Specifies magnitudes that are given in tabular format. Some keywords require magnitudes to be specified as a load curve. Should a constant be needed, it may be represented as a curve by repeating its value for time steps 0 and 1.

Card1

- LCID = ID for load curve, is incremented every time a new load curve is defined.

Cards 2, 3, 4...

- A = abscissa value, usually time.
- O = ordinate (function) value.

*DEFINE_CURVE_FUNCTION

Specifies a time function where the magnitude is defined by a function expression.

Card1

- LCID = ID for load curve, is incremented every time a new load curve is defined.

Cards 2, 3 , 4

Lines of eighty characters used to form the text of the function.

5.16. Acceleration and Gravity

*LOAD_BODY_X

Specifies gravitational or other acceleration loads in the x direction. The load is applied to all nodes in the model.

Card

- LCID = ID of the load curve that represents the magnitude of the load (see *DEFINE_CURVE (p. 204)).
- SF = 1.0 (default), load curve scale factor.
- LCIDDR = 0 (default), ID of load curve defined for dynamic relaxation.
- XC = 0.0 (default), X-center of rotation needed for angular velocities.
- YC = 0.0 (default), Y-center of rotation needed for angular velocities.
- ZC = 0.0 (default), Z-center of rotation needed for angular velocities.
- CID = ID of local coordinate system used. Set to 0 for the global coordinate system.

*LOAD_BODY_Y

Specifies gravitational or other acceleration loads in the y direction. The load is applied to all nodes in the model.

Card

(see *LOAD_BODY_X).

*LOAD_BODY_Z

Specifies gravitational or other acceleration loads in the z direction. The load is applied to all nodes in the model.

Card

(see *LOAD_BODY_X).

5.17. Supports

*BOUNDARY_SPC_SET

Specifies Fixed Support, Simply Supported, and Fixed Rotation constraints.

Card

- NSID = ID of set of nodes to which the boundary is applied.
- CID = ID of the associated coordinate system. 0 specifies the global coordinate system.
- DOFX = 0 or 1. 0 means that the translation is free and 1 that the translation is constrained along the x direction. It is set to 0 for the Fixed Rotation boundary condition and to 1 for the Simply Supported boundary condition.
- DOFY = 0 or 1. 0 means that the translation is free and 1 that the translation is constrained along the y direction. It is set to 0 for the Fixed Rotation boundary condition and to 1 for the Simply Supported boundary condition.
- DOFZ = 0 or 1. 0 means that the translation is free and 1 that the translation is constrained along the z direction. It is set to 0 for the Fixed Rotation boundary condition and to 1 for the Simply Supported boundary condition.
- DOFRX = 0 or 1. 0 means that the rotation is free and 1 that the rotation is constrained along the x direction. It is set to 0 for the Simply Supported Boundary Condition and to 1 for the Fixed Rotation Boundary Condition when the degree of freedom is fixed in that direction.
- DOFRY = 0 or 1. 0 means that the rotation is free and 1 that the rotation is constrained along the y direction. It is set to 0 for the Simply Supported Boundary Condition and to 1 for the Fixed Rotation Boundary Condition when the degree of freedom is fixed in that direction.
- DOFRZ = 0 or 1. 0 means that the rotation is free and 1 that the rotation is constrained along the z direction. It is set to 0 for the Simply Supported Boundary Condition and to 1 for the Fixed Rotation Boundary Condition when the degree of freedom is fixed in that direction.

*BOUNDARY_PRESCRIBED_MOTION_RIGID

See *BOUNDARY_PRESCRIBED_MOTION_SET

*BOUNDARY_PRESCRIBED_MOTION_SET

Specifies velocity and displacement boundary conditions.

Card1

- ID = ID of set of nodes or part (for rigid bodies) to which the boundary condition is applied.

- DOF = 1, 2 or 3 depending on whether the boundary condition is in the x, y or z direction respectively, and is a translational boundary condition.

  DOF = 4, 5 or 6 depending on whether the boundary condition is in the x, y or z direction respectively, and is a rotational boundary condition.

  Setting 4 is used if the boundary is applied according to a local coordinate system.

- VAD = 0 or 2 depending whether the boundary condition is velocity or displacement.
- LCID = ID of the curve prescribing the magnitude of the boundary condition. Constant values of velocity are applied as a step function from time = 0. Constant values of displacement are ramped from zero at time = 0 to the constant value at termination time. This is done to make sure that displacements are applied in a transient fashion.
- SF = 1.0 (default) scale factor for load curve.
- VID = 0 (default). ID of vector that defines the local coordinate system the boundary condition is applied with.
- DEATH = 0.0 (default), sets it to 1E28.
- BIRTH = 0, the motion is applied from the beginning of the solution.

Card2: not required.

*BOUNDARY_TEMPERATURE

This keyword is written when a Temperature object is inserted into the system. It is always written with the option _SET. Even with constant values, a *DEFINE_CURVE keyword is used.

Card:

- NID: Automatically written.
- TLCID: Automatically written. The corresponding *DEFINE_CURVE keyword is written using the values written in **Magnitude** under **Definition**.
- TMULT = 1.

*BOUNDARY_CONVECTION

This keyword is written when a Convection object is inserted into the system. It is always written with the option _SET. Even with constant values, a *DEFINE_CURVE keyword is used.

Card:

- SSID: Automatically written.
- HLCID: Automatically written. The corresponding *DEFINE_CURVE keyword is written using the values written in **Film Coefficient** under **Definition**.
- HMULT = 1.

- TLCID: Automatically written. The corresponding *DEFINE_CURVE keyword is written using the values written in **Ambient Temperature** under **Definition**.
- TMULT = 1.

*BOUNDARY_FLUX

This keyword is written when a Heat Flux object is inserted into the system. It is always written with the option _SET. Even with constant values, a *DEFINE_CURVE keyword is used.

Card:

- SSID: Automatically written.
- LCID: Automatically written. The corresponding *DEFINE_CURVE keyword is written using the values written in **Magnitude** under **Definition**.
- MLC1 = -1
- MLC2 = -1
- MLC3 = -1
- MLC4 = -1

5.18. Loads

*LOAD_NODE_SET

Applies a concentrated nodal force to a set of nodes.

Card

- NSID = the set of nodes where the force is applied.
- DOF = 1, 2 or 3 depending on the force direction x, y or z.
- LCID = ID of the load curve that describes the magnitude of the force (see *DEFINE_CURVE (p. 204)).
- SF = 1.0 (default), load curve scale factor.
- CID = ID of local coordinate system used. Set to 0 for the global coordinate system.

*LOAD_RIGID_BODY

Applies a concentrated nodal force to a rigid body. The force is applied at the center of mass.

Card

See *LOAD_NODE_SET. Note that parameter NSID is replaced by PID which is the ID of the part the force is applied to.

*LOAD_SEGMENT_SET

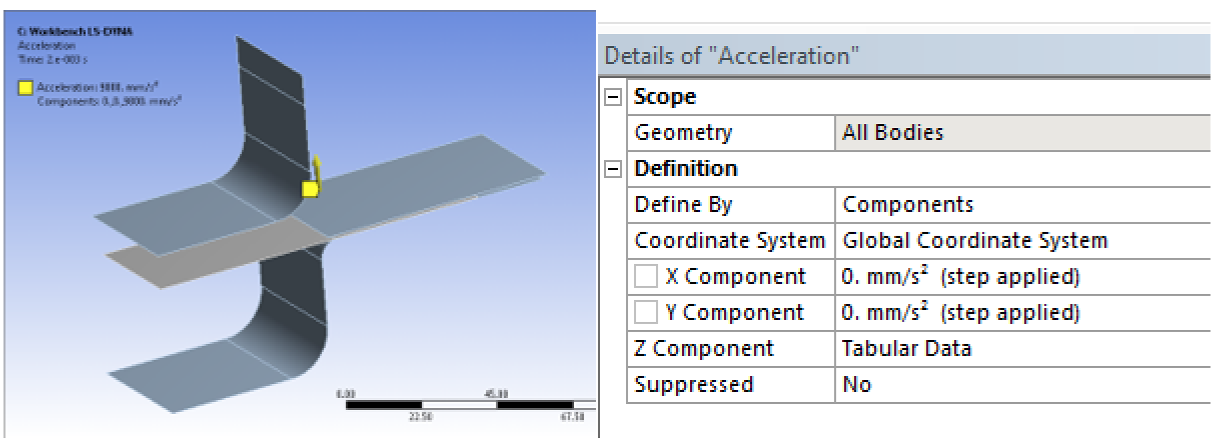
Applies a distributed pressure load over each segment in a segment set.

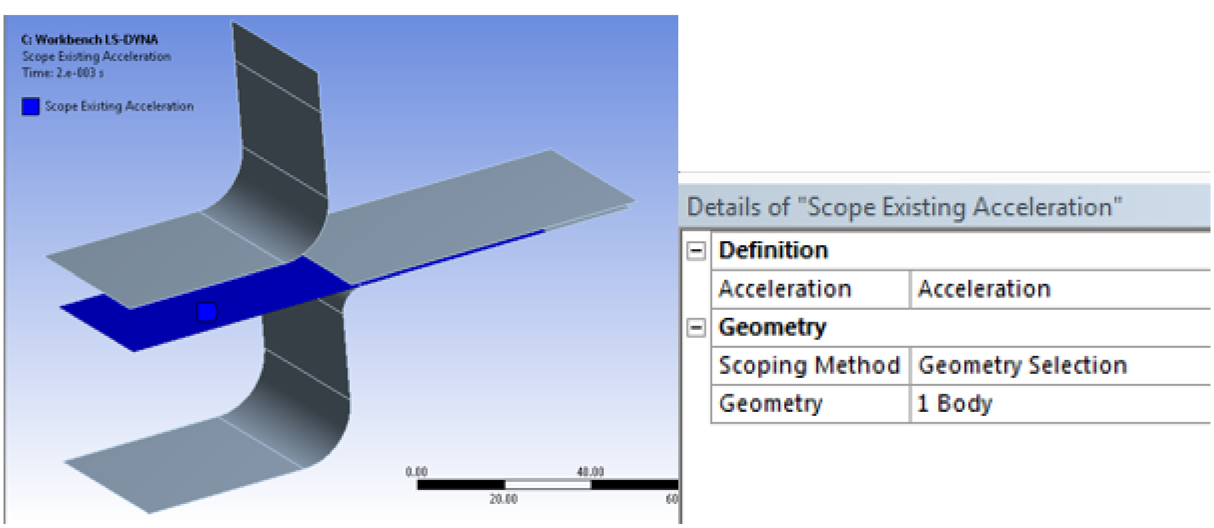
Card

- LCID = ID of the load curve that describes the magnitude of the pressure (see *DEFINE_CURVE (p. 204)).
- SSID = ID of set of nodes to which the pressure is applied.
- SF = 1.0 (default), load curve scale factor.
- AT = arrival time for pressure is assigned the time at load step 1 if pressure is given in tabular form or 0 if constant pressure.

*LOAD_BODY_PARTS

By Default Acceleration or Gravity load are applied to all bodies in the model, and it is sometimes convenient to apply these loads to only a portion of the model. *LOAD_BODY_PARTS is used in that context.

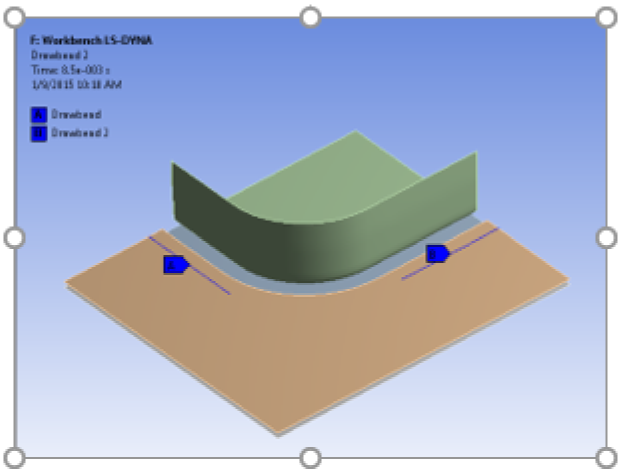




*DEFINE_BOX_DRAWBEAD

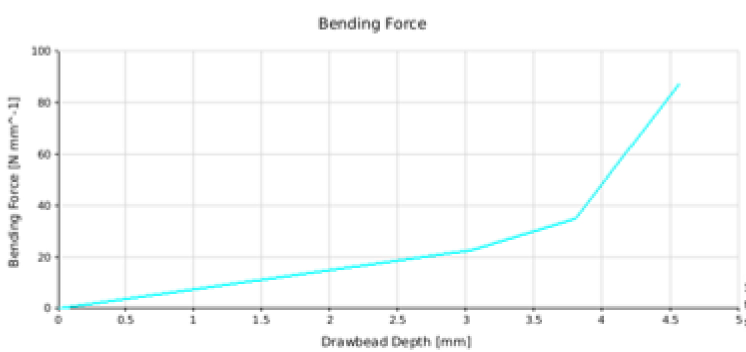
*CONTACT_DRAWBEAD_ID

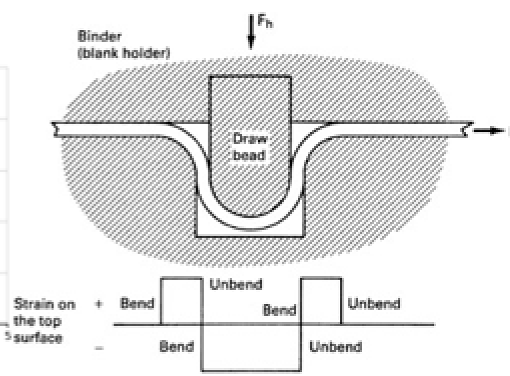
These two keywords are used to support Drawbead modeling. Drawbeads are used to control the flow of sheet metal into the die cavity during stretch-draw forming of large panels. LS-DYNA enables them to be defined them through a specific contact algorithm.



Details of "Drawbead"

Draw Bead Lines	
Scoping Method	Geometry Selection
Geometry	2 Bodies
Blank	
Scoping Method	Geometry Selection
Geometry	1 Body
Die	
Scoping Method	Geometry Selection
Geometry	1 Body
Tooling Movement	
Coordinate System	Global Coordinate System
Direction	tooling moves in z-direction
Definition	
Friction	0.1
Radius	5 mm
Drawbead Depth	4.5 mm
Number Of Integration Points along the draw bead	0
Bending Force	Tabular Data





*DEFINE_BOX

Define a box-shaped volume. Two diagonally opposite corner points of a box are specified in global or local coordinates. The box volume is then used for various specifications for a variety of input options, such as velocities, displacement, and so forth.

Details of "Box_X"

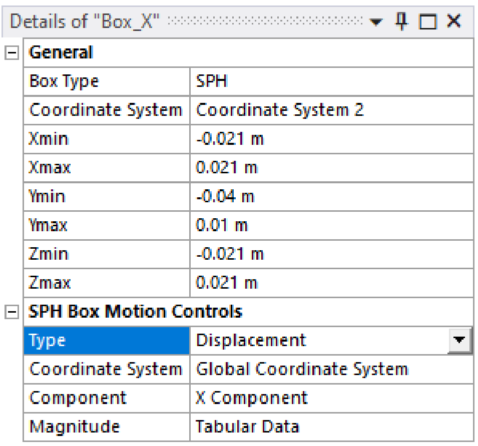
General	
Box Type	Standard
Coordinate System	Coordinate System 2
Xmin	-0.021 m
Xmax	0.021 m
Ymin	-0.04 m
Ymax	0.01 m
Zmin	-0.021 m
Zmax	0.021 m

Card

- Box ID = ID of the Box object
- XMN = Xmin
- XMX = Xmax
- YMN = Ymin
- YMX = Ymax
- ZMN = Zmin
- ZMX = Zmax

*DEFINE_BOX_SPH

Define a box-shaped volume. Two diagonally opposite corner points of a box are specified in global coordinates. Particle approximations of SPH elements are computed when particles are located inside the box. The load curve describes the box motion as function of time.



Card 1

- Box ID = ID of the Box object
- XMN = Xmin
- XMX = Xmax
- YMN = Ymin
- YMX = Ymax
- ZMN = Zmin
- ZMX = Zmax

Card 2

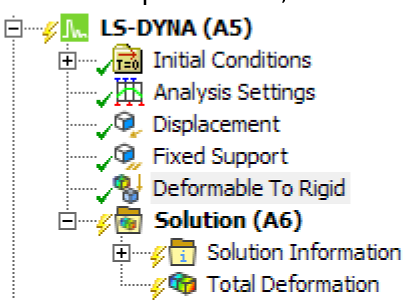
- VID = Vector ID created from **Component** using *DEFINE_VECTOR (p. 190)
- LCID = Load curve ID created from **Magnitude** tabular data using *DEFINE_CURVE (p. 204)

- VD =
  - 0 if Type = Velocity
  - 1 if Type = Displacement

*DEFORMABLE_TO_RIGID

In some dynamic applications, long duration, large rigid body motions arise that are prohibitively expensive to simulate if the majority of elements in the model are deformable. One such example would be an automotive rollover where the time duration of the rollover would dominate the CPU cost relative to the impact that occurs much later. To improve efficiency for this class of applications, The LS-DYNA system offers the capability to switch a subset of materials from a deformable state to a rigid state, and then back to deformable. By switching the deformable parts to rigid during the rigid body motion stage, you can achieve significant savings in computation time.

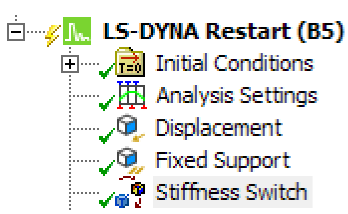
Deformable/rigid switching is inherently related to performing a restart. You need to stop the analysis, define the part switch, and then restart the analysis again.

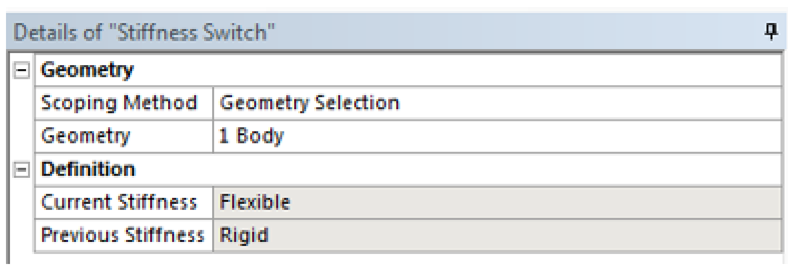


The scoped body is defined with flexible material properties. It is switched to rigid for the first calculation.

*RIGID_DEFORMABLE_D2R

The body stiffness can then be modified in dependent calculations. If the stiffness was rigid, it is switched to flexible



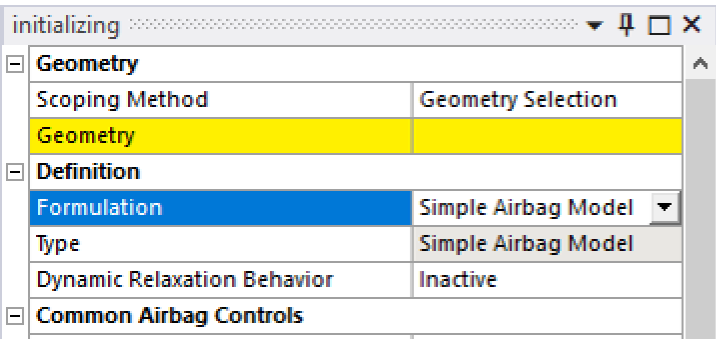


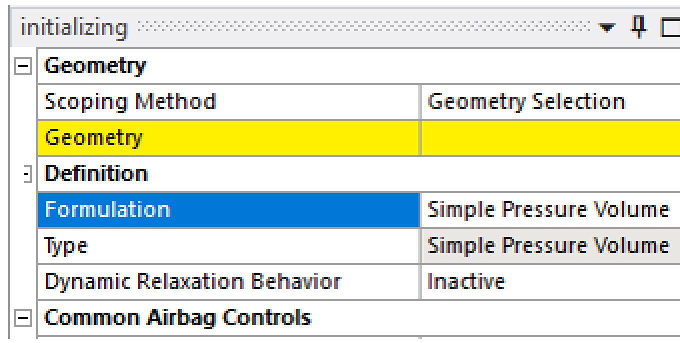
*AIRBAG_SIMPLE_AIRBAG_MODEL

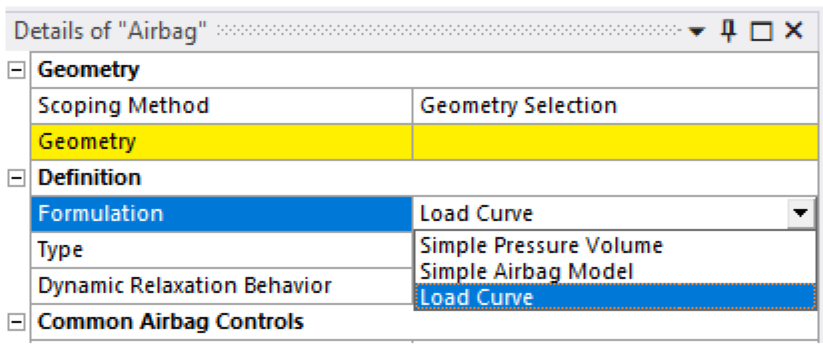
*AIRBAG_SIMPLE_PRESSURE_VOLUME

*AIRBAG_LOAD_CURVE

These three keywords are used to support airbag modeling. Set the type to Simple Airbag Model, Simple Pressure Volume, or Load Curve, respectively. See Airbag or Simple Pressure Volume (p. 52) for more information about using the airbag object.



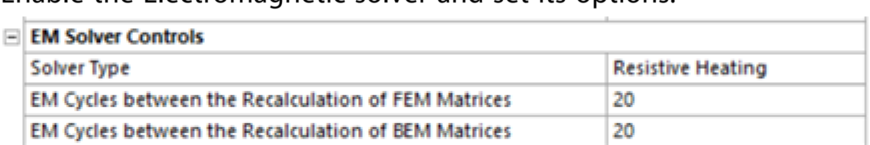




5.19. Electromagnetic Support

*EM_CONTROL

Enable the Electromagnetic solver and set its options.



Card

- emsol = **Solver Type** under the EM Solver Controls section of the Analysis Settings object. Currently Resistive Heating is the only option.
- ncylfem = **EM Cycles between the Recalculation of FEM Matrices** under the EM Solver Controls section of the Analysis Settings object.

- ncylbem = **EM Cycles between the Recalculation of BEM Matrices** under the EM Solver Controls section of the Analysis Settings object.

*EM_CONTROL_TIMESTEP

EM Step Controls	
Time Step Type	Automatic
Factor to Automatic Time Step	1
Minimum Time Step	0 s
Maximum Time Step	1 s
Decrease the Solid Mechanics Time Step to match EM Time Step	Yes

Controls the EM time step and its evolution.

Card

- tstype is set by **Step Type** under the EM Step Controls section of the Analysis Settings object:
  - = 1 when Step Type = Constant
  - = 3 when Step Type = Automatic
  - = 4 when Step Type = Controlled by Thermal Solver
- dtconst = **Time Step Size** under the EM Step Controls section of the Analysis Settings object when Step Type is set to Constant.
- factor = **Factor to Automatic Time Step** under the EM Step Controls section of the Analysis Settings object when Step Type is set to Automatic.
- tsmin = **Minimum Time Step** under the EM Step Controls section of the Analysis Settings object when Step Type is set to Automatic or Controlled by Thermal Solver.
- tsmax = **Maximum Time Step** under the EM Step Controls section of the Analysis Settings object when Step Type is set to Automatic or Controlled by Thermal Solver..
- mecats =
  - 0 when **Decrease the Solid Mechanics Time Step to match EM Time Step** under the EM Step Controls section of the Analysis Settings is set to No.

    The EM time step will go below the solid mechanics timestep, and several EM solves will occur between two solid mechanics time steps to ensure time consistency.
  - 1 when **Decrease the Solid Mechanics Time Step to match EM Time Step** under the EM Step Controls section of the Analysis Settings is set to Yes

    The solid mechanics time step will adapt and decrease to the EM time step value so that only one EM solve occurs between two solid mechanics solves.

5.20. Discrete Connections

*SECTION_DISCRETE

Defines section properties for solid elements. DRO is the Displacement/Rotation Option.

- SECID = ID of the section.
- DRO =
  - 0 for translational spring/damper.
  - 1 for torsional spring/damper.

*ELEMENT_DISCRETE

Specifies spring elements.

- EID = ID of the element.
- PID = ID of the part it belongs to.
- N1 = ID of nodal point 1.
- N2 = ID of nodal point 2.

*MAT_SPRING_ELASTIC

This keyword is used in support of spring connections, the K parameter of this material keyword is the stiffness of the string.

- MID = ID of material type. Must be unique between the material keyword definitions.
- K = Elastic stiffness (force/displacement) or (moment/rotation).

*MAT_DAMPER_VISCOUS

This keyword is used in support of spring connections. The damping constant DC of this material is the damping parameter of the spring, if this damping is non null.

- MID = ID of material type. Must be unique between the material keyword definitions.
- DC = Damping constant (force/displacement rate) or (moment/rotation rate).

5.21. Other Supports

*BOUNDARY_NON_REFLECTING

Specifies impedance boundaries. Impedance boundaries can only be applied on solid elements in LS-DYNA.

Card

- SSID = ID of segment on whose nodes the boundary is applied (see *SET_SEGMENT bellow).
- AD = 0.0 (default) for setting the activation flag for dilatational waves to on.
- AS = 0.0 (default) for setting the activation flag for shear waves to on.

*BOUNDARY_SLIDING_PLANE

Defines a boundary plane for sliding symmetry.

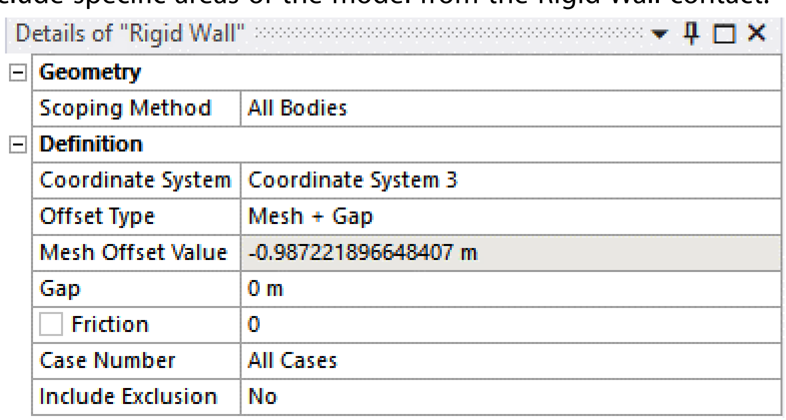
Note:

The normal is defined as the z-axis vector of the coordinate system defined in Mechanical.

- NSID = ID of the set of nodes to which the boundary is applied
- VX = X component of vector defining normal.
- VY = Y component of vector defining normal.
- VZ = Z component of vector defining normal.
- COPT =
  - 0 if the Option is set to node moves on normal plane.
  - 1 if the Option is set to node moves only in vector direction.

*RIGID_WALL_PLANAR

The Rigid Wall object provides a simple way of treating contact between a rigid surface and nodal points of a deformable body called contact nodes. The Rigid Wall can be scoped to Geometry or Named Selections. You can exclude specific areas of the model from the Rigid Wall contact.



The **Offset Type** property can be set under the Rigid Wall object. This property provides the option to specify an offset for the rigid wall plane along the normal Z direction of the selected coordinate system.

Offset Type Options	Description
None	The default value of the rigid wall plane is positioned at Z = 0 in the selected coordinate system.
Mesh	The rigid wall plane is positioned at Z = Mesh Offset Value in the selected coordinate system, where Mesh Offset Value is calculated as the distance delta Z in the negative Z direction to the furthest mesh node of the scoped geometry.
Gap	The rigid wall plane is positioned at Z = -Gap in the selected coordinate system, where Gap is a user-defined delta Z applied in the selected coordinate system.
Mesh + Gap	The rigid wall plane is positioned at Z = Mesh Offset Value - Gap in the selected coordinate system, where Mesh Offset Value is calculated as the distance delta Z in the negative Z direction to the furthest mesh node of the scoped geometry and Gap is a user-defined delta Z applied in the selected coordinate system.

Note:

A valid mesh must be generated for offset types of Mesh or Mesh + Gap, otherwise the property is shown as invalid, and no offset is applied.

- NSID = ID of the set of nodes to which the boundary is applied.
- XT, YT, ZT, XH, YH, ZH are calculated from the Coordinate System definition; the normal Z of the coordinate system is the normal to the plane.
- FRIC = Friction.

5.22. Environment Temperature

*INITIAL_TEMPERATURE_SET

This keyword is added in coupled structural thermal analyses, where it defines the initial temperature of the environment.

- NSID = 0. All nodes of the model are initialized with the temperature Temp.
- Temp = temperature of the environment.

5.23. ASCII Files

The following keywords specify time-history output (ASCII format) for an explicit dynamics analysis. The time history files output are requested through **Time History Output Controls** section of the **Analysis Settings** . Up to 10 ASCII files can be requested in the GUI. The sampling time is calculated from the number of values requested and from the end time of the simulation.

Time History Output Controls	
Calculate Results At	Equally Spaced Points
--- Value	1000
Output	Boundary Condition Forces and Energy
Output	Nodal Data
Output	Wall Forces
Output	Discrete Elements Data
Output	Material energies Data
Output	Nodal Interface Forces
Output	Resultant Interface Force Data
Output	Deformed Geometry Data
Output	Spc Reaction Forces Data
Output	Nodal Constraint Reaction Force Data (spotwelds ...

The results files (`GLSTAT` , `BNDOUT` , `RCFORC` , `SPCFORC` , `MATSUM`) are always output by LS-DYNA. The sampling frequency can, however, be modified.

*DATABASE_HISTORY_NODE_SET

Controls which nodes or elements are output into the binary history file and the ASCII file `NODOUT` for a particular result tracker.

Card

ID1 = the ID of the component defined by the result tracker object.

*DATABASE_BNDOUT

Specifies the sampling parameters for the `BNDOUT` results file (stores Boundary condition forces and energy).

Card

DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_DEFGEO

Specifies the sampling parameters for the `DEFGEO` results file (stores Deformed Geometry Data).

Card

DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_DEFORC

Specifies the sampling parameters for the `DEFORC` results file (Discrete Elements Data).

Card

DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_GCEOUT

Specifies the sampling parameters for the `GCEOUT` results file (Geometric Contact Entities).

Card

DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_GLSTAT

Specifies the sampling parameters for the `GLSTAT` results file (stores general energy results).

Card

- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_INTFORC

Specifies the sampling parameters for the `JNTFORC` results file (stores Joint Forces).

DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_MATSUM

Specifies the sampling parameters for the `MATSUM` results file (stores general energy and velocity results as the GLSTAT file but it stores them per body. It is necessary for rigid bodies).

Card

- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_NCFORC

Specifies the sampling parameters for the `NCFORC` results file (stores Nodal Interface Forces).

Card

- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_NODOUT

Specifies the sampling parameters for the `NODOUT` results file (stores displacement and velocity results).

Card

- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_RBDOUT

Specifies the sampling parameters for the RBDOUT results file (stores Rigid Body Data).

Card

DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_RCFORC

Specifies the sampling parameters for the RCFORC results file (stores contact forces).

Card

DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_RWFORC

Specifies the sampling parameters for the RWFORC results file (stores Rigid Wall forces).

Card

DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_SLEOUT

Specifies the sampling parameters for the SLEOUT results file (stores sliding interface forces).

Card

- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_SPCFORC

Specifies the sampling parameters for the SPCFORC results file (stores reaction forces).

Card

- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

*DATABASE_SWFORC

Specifies the sampling parameters for the `SWFORC` results file (stores the spotweld and rivet forces).

Card

- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings**.

5.24. Database Output Settings

*DATABASE_EXTENT_BINARY

Control to some extent the content of binary output databases d3plot, d3thdt, and d3part. Four parameters are set by LS-DYNA:

Card1

- STRFLG: Flag for including the strain tensor for shells.
- SIGFLG: Flag for including the stress tensor for shells.
- EPSFLG: Flag for including the effective plastic strains for shells.

Card3

- MSSCL: Output nodal information related to mass scaling into the d3plot database.
- THERM = 2 when **Solver Type** under **Analysis Settings** is set to Coupled Structural Thermal Analysis; 0 otherwise.

*DATABASE_BINARY_D3PLOT

Specifies the sampling parameters for the binary D3PLOT results plotting file.

Card

- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings** if **Calculate Results At** is set to Equally Spaced Time Points (this value defaults to 20).

*DATABASE_BINARY_INTFOR

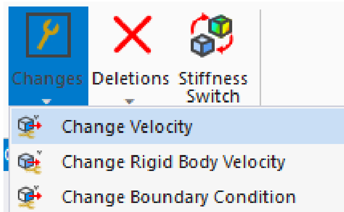
Specifies the sampling parameters for the binary intfor results file. This file contains contact information (pressure, nodal contact forces).

Card

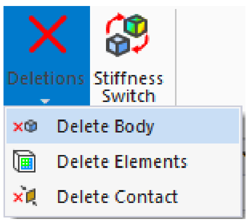
- DT = **End Time** divided by **Value** from the **Time History Output Controls** section of the **Analysis Settings** if **Calculate Results At** is set to Equally Spaced Time Points (this value defaults to 20).

5.25. Restart

You can change the velocity for rigid bodies or other parts of the model using the options from the Change icon in the LSDYNA Small Restart ribbon tab.



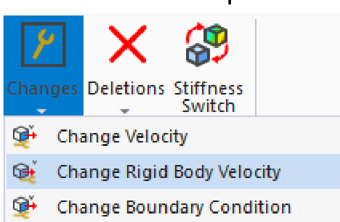
You can also delete parts of the model during a small restart using the options from the Deletions icon in the LSDYNA Small Restart ribbon tab



5.25.1. Changing Velocity

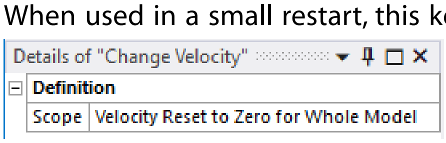
*CHANGE_VELOCITY_RIGID_BODY

When used in a small restart, this keyword changes the velocity of a rigid body. If you don't change the velocity of rigid body, its velocity at the beginning of the restart will be the same as the velocity at the end of the previous analysis.



*CHANGE_VELOCITY_ZERO

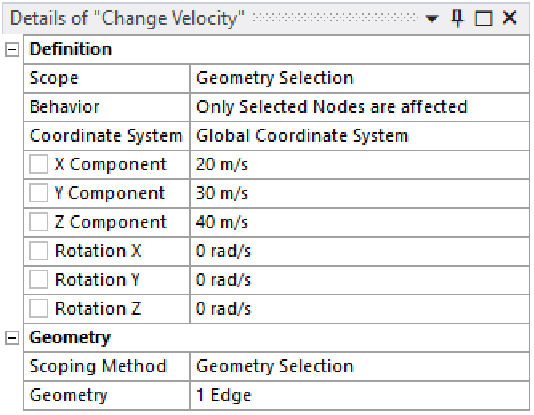
When used in a small restart, this keyword resets the velocity of the nodes for the whole model.



This keyword is used when a **Change Velocity** boundary condition is used, and the scope is set to Velocity reset to zero for Whole Model.

*CHANGE_VELOCITY_ONLY

When used in a small restart, this keyword changes the velocity of the specified nodes. It only affects the velocity of selected nodes. If you don't change the velocity of the nodes, their velocity at the beginning of the restart will be the same as the velocity at the end of the previous analysis.

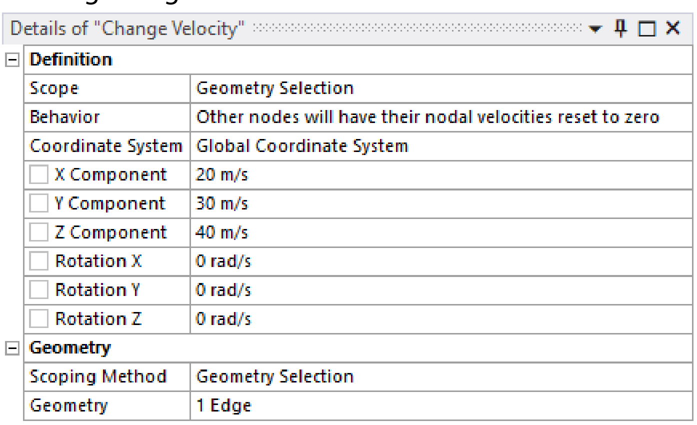


This keyword is used when a **Change Velocity** boundary condition is used and the **Scope** is set to Geometry Selection and the **Behavior** to *Only Selected Nodes are affected*.

A coordinate system can be selected for the change velocity object, and it can be scoped to vertices, edges, faces, or bodies.

*CHANGE_VELOCITY

When used in a small restart, this keyword changes the velocity of the specified nodes. The other nodes have their velocity set to zero. If you don't change the velocity of the nodes, their velocity at the beginning of the restart will be the same as the velocity at the end of the previous analysis.



This keyword is used when a **Change Velocity** boundary condition is used and the **Scope** is set to Geometry Selection and the **Behavior** to *Other nodes will have their nodal velocities reset to zero*.

Note:

When the Change Velocity object is scoped to a body, there will be no **Behavior** field and the *CHANGE_VELOCITY_GENERATION (p. 195) keyword is written.

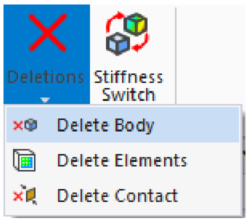
A coordinate system can be selected for the change velocity object, and it can be scoped to vertices, edges, faces, or bodies.

5.25.2. Delete Model Pieces

*DELETE_PART

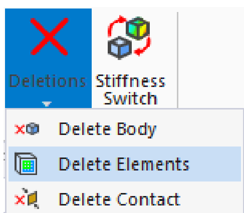
In an explicit dynamic small restart analysis, this keyword can be used to unselect a part during the solution even if it is referenced in some way (such as in a contact definition).

This keyword is used when a Delete Body object is used.



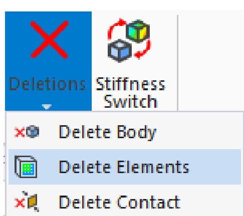
*DELETE_ELEMENT_BEAM

This keyword is used when a Delete Elements object is used and the scoped elements contain beam elements.



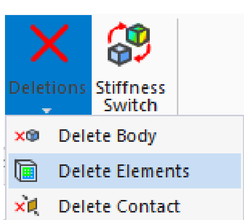
*DELETE_ELEMENT_SHELL

This keyword is used when a Delete Elements object is used and the scoped elements contain shell elements.



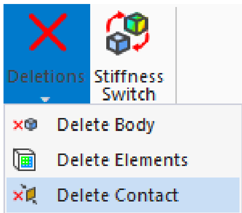
*DELETE_ELEMENT_SOLID

This keyword is used when a Delete Elements object is used and the scoped elements contain solid elements.



*DELETE_CONTACT

This keyword allows you to delete a particular contact specification that was defined. The delete option permanently deletes the contact from the database.



5.26. End of Input File

*END

Terminates the keyword file. It has no parameter cards.

Chapter 6: Material Models Available in Workbench

Engineering Data in Workbench gives you access to many of the material models available for LS-DYNA. The sections below direct you to the discussion of the materials in the Mechanical and Explicit Dynamics documentation and tell you what LS-DYNA keyword is used when you specify a certain material.

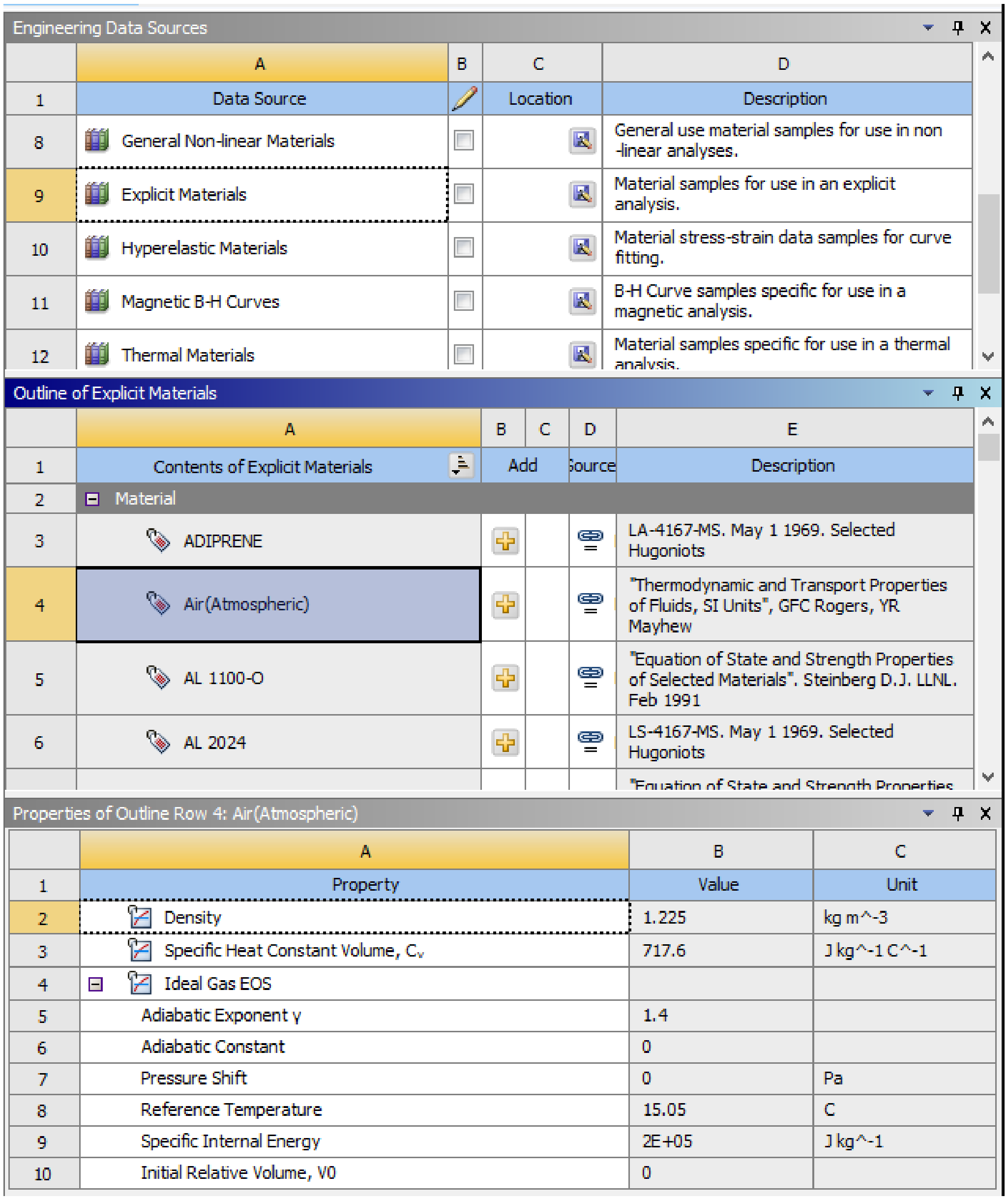
6.1. Introduction

LS-DYNA supports a large material library and can, therefore, simulate nearly any application. LS-DYNA materials offer many features including:

- Strain rate dependent plasticity models with strain failure criterion.
- Temperature dependent and temperature sensitive plasticity models.
- Equations of state and null material models (bird-strike analyses, etc.).

The modeling of such phenomena can generally be broken down into three components:

The LS-DYNA system provides access to part of the Explicit Materials library, including Air (Atmospheric) which uses the Ideal Gas EOS (p. 248), and most of the Explosive materials which use the Explosive JWL (p. 248); for example TNT, C4, COMP B, HMX, H6, etc.



Note:

Ideal Gas EOS (p. 248) and Explosive JWL (p. 248) do not have the same fields as the equivalent material properties shown in LS-DYNA External Model Material Properties (p. 245) because they show the fields that are contained in the original Autodyn material library. Those fields are converted into the LS-DYNA keyword parameters as described in the Explicit Dynamics Migration Guide.

6.1.1. Equation of State

An equation of state describes the hydrodynamic response of a material.

This is the primary response for gases and liquids, which can sustain no shear. Their response to dynamic loading is assumed hydrodynamic, with pressure varying as a function of density and internal energy.

This is also the primary response for solids at high deformation rates, when the hydrodynamic pressure is far greater than the yield stress of the material.

6.1.2. Material Strength Model

Solid materials may initially respond elastically, but under highly dynamic loadings, they can reach stress states that exceed their yield stress and deform plastically. Material strength laws describe this non-linear elastic-plastic response.

6.1.3. Material Failure Model

Solids usually fail under extreme loading conditions, resulting in crushed or cracked material. Material failure models simulate the various ways in which materials fail. Liquids will also fail in tension, a phenomenon usually referred to as cavitation.

6.2. Density

See the description in Density .

6.3. Linear Elastic

See the description in Linear Elastic .

6.3.1. Isotropic Elasticity

See the description in Isotropic Elasticity .

This material behavior is written as *MAT_ELASTIC (p. 170) .

6.3.2. Orthotropic Elasticity

See the description in Orthotropic Elasticity .

This material behavior is written as *MAT_ORTHOTROPIC_ELASTIC (p. 178) .

6.3.3. Anisotropic Elasticity

This material description requires the full elasticity matrix. Because of symmetry, only 21 constants are required:

$$C = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ & & C_{33} & C_{34} & C_{35} & C_{36} \\ & & & C_{44} & C_{45} & C_{46} \\ \text{Symm} & & & & C_{55} & C_{56} \\ & & & & & C_{66} \end{bmatrix}$$

This material is only valid for solid elements. Material properties are locally orthotropic with material axes defined based on the mesh orientation defined in the Mechanical application (Body orientation).

6.4. Test Data

See the description in Test Data .

6.5. Hyperelasticity

Following are several forms of strain energy potential (Ψ) provided for the simulation of nearly incompressible hyperelastic materials. The different models are generally applicable over different ranges of strain as illustrated in Hyperelasticity , however these numbers are not definitive and users should verify the applicability of the model chosen prior to use.

Hyperelastic materials may only be used in solid and shell elements for LS-DYNA simulations.

See Hyperelasticity for additional information on the models discussed in this section.

6.5.1. Blatz-Ko Hyperelasticity

Blatz-Ko materials are only for rubber materials under compression. Poisson's ratio (NUXY) is automatically set to 0.463 by LS-DYNA, so only density and initial shear modulus are required. This model uses the second Piola-Kirchhoff stress:

$$S_{ij} = G\left[\frac{1}{V}C_{ij} - V^{\left(\frac{1}{1-2\nu}\right)}\delta_{ij}\right]$$

G is the shear modulus, V is the relative volume, ν is the Poisson's ratio, C_{ij} is the right Cauchy-Green strain tensor, and δ_{ij} is the Kronecker delta.

This material behavior is written as *MAT_BLATZ-KO_RUBBER (p. 170) .

6.5.2. Mooney-Rivlin

The 2, 3, 5 parameter Mooney-Rivlin hyperelastic material models have been implemented. The 9 parameter version of this material model is not supported.

This material behavior is written as *MAT_HYPERELASTIC_RUBBER (p. 174) .

6.5.3. Polynomial

This material behavior is written as *MAT_HYPERELASTIC _RUBBER (p. 174) .

6.5.4. Yeoh

The first order and second order of this material model have been implemented.

This material behavior is written as *MAT_HYPERELASTIC_RUBBER (p. 174) .

6.5.5. Ogden

This material behavior is written as *MAT_OGDEN_RUBBER (p. 178) .

6.6. Plasticity

All stress-strain input should be in terms of true stress and true (or logarithmic) strain and result in all output as also true stress and true strain. For small-strain regions of response, true stress-strain and engineering stress-strain are approximately equal. If your stress-strain data is in the form of engineering stress and engineering strain you can convert:

- **strain** from engineering strain to logarithmic strain using: $c_{ln}=ln\left(1+c_{eng}\right)$
- **engineering stress** to true stress using: $\sigma_{true}=\sigma_{eng}\left(1+c_{eng}\right)$

Note:

This stress conversion is only valid for incompressible materials.

The following Plasticity models are discussed in this section:

6.6.1. Bilinear Isotropic Hardening

This plasticity material model is often used in large strain analyses. A bilinear stress-strain curve requires that you input the **Yield Strength** and **Tangent Modulus** . The slope of the first segment in the curve is equivalent to the Young's modulus of the material while the slope of the second segment is the tangent modulus.

This material behavior is written as *MAT_PLASTIC_KINEMATIC (p. 181) . The parameter beta of this keyword is set to 1.

Custom results variables available for this model:

Name	Description	Solids	Shells	Beams
EPS	Effective Plastic Strain	Yes	Yes*	No

*Resultant value over shell/beam section.

6.6.2. Multilinear Isotropic Hardening

This plasticity material model is often used in large strain analyses. Do not use this model for cyclic or highly nonproportional load histories in small-strain analyses.

You must supply the data in the form of plastic strain vs. stress. The first point of the curve must be the yield point, that is, zero plastic strain and yield stress. The slope of the stress-strain curve is assumed to be zero beyond the last user-defined stress-strain data point. No segment of the curve can have a slope of less than zero.

Note:

You can define up to 10 stress strain pairs using this model in explicit dynamics systems. Temperature dependence of the curves is not directly supported. Temperature dependent plasticity can be represented using the Johnson-Cook plasticity model.

Custom results variables available for this model:

Name	Description	Solids	Shells	Beams
EPS	Effective Plastic Strain	Yes	Yes*	No

*Resultant value over shell/beam section.

6.6.3. Bilinear Kinematic Hardening

This plasticity material model assumes that the total stress range is equal to twice the yield stress, to include the Bauschinger effect. This model may be used for materials that obey Von Mises yield criteria (includes most metals). The tangent modulus cannot be less than zero or greater than the elastic modulus.

This material behavior is written as *MAT_PLASTIC_KINEMATIC (p. 181) . The parameter beta of this keyword is set to 0.

Custom results variables available for this model:

Name	Description	Solids	Shells	Beams
EPS	Effective Plastic Strain	Yes	Yes*	No

*Resultant value over shell/beam section.

6.6.4. Johnson-Cook Strength

See Johnson-Cook Strength for information about this model. However, the strain rate correction parameter described in this material model is not used by LS-DYNA.

This material behavior is written as *MAT_JOHNSON_COOK (p. 174) or *MAT_SIMPLIFIED_JOHNSON_COOK (p. 184) depending on whether it is used in combination with an equation of state or not. The simplified form is used when no equation of state is defined. The thermal terms are discarded in that scenario.

Custom results variables available for this model:

Name	Description	Solids	Shells	Beams
EPS	Effective Plastic Strain	Yes	Yes*	No
TEMP	Temperature**	Yes	Yes*	No

*Resultant value over shell/beam section.

**Temperature will be non-zero only if a specific heat capacity is defined.

6.6.5. Cowper-Symonds Power Law Hardening

The Cowper-Symonds power law hardening lets you define plastic behavior with bilinear isotropic hardening and power law hardening defined with strength coefficient K and hardening coefficient n. Strain rate effects are accounted for by the Cowper-Symonds strain rate parameters, C and P.

Yield surface can be scaled for strain rate dependence or the latter can be defined using a fully visco-plastic formation.

This material behavior is written as *MAT_POWER_LAW_PLASTICITY (p. 182) .

Name	Symbol	Units	Notes
Initial Yield Stress	A	Stress	
Hardening Constant	K	Stress	
Hardening Exponent	n	None	
Strain Rate Constant	C	None	Assumed 1/second in all cases
Strain Rate Constant	P	None	
Strain Rate Correction	-	None	Option List: Scale Yield Stress Viscoplastic

Custom results variables available for this model:

Name	Description	Solids	Shells	Beams
EPS	Effective Plastic Strain	Yes	Yes*	Yes*

*Resultant value over shell/beam section.

6.6.6. Rate Sensitive Power Law Hardening

Strain rate dependent plasticity model typically used for superplastic forming analyses. The material model follows a Ramburgh-Osgood constitutive relationship of the form:

$$\sigma_y = k\varepsilon^m \dot{\varepsilon}^n$$

where:

k is the material coefficient,

m is the hardening coefficient,

n is the strain rate parameter.

Name	Symbol	Units	Notes
Hardening Constant	K	Stress	
Hardening Exponent	m	None	
Strain Rate Constant	n	None	
Reference Strain Rate		None	Units fixed at 1/sec Default = 0.0002

This material behavior is written as *MAT_RATE_SENSITIVE_POWERLAW_PLASTICITY (p. 182) .

6.6.7. Cowper-Symonds Piecewise Linear Hardening

This model is very efficient in solution and is most commonly used in crash simulations. It is similar to the multilinear isotropic hardening behavior. Stress-strain behavior is defined with a load curve of effective true stress versus effective plastic true strain. Failure strain can be input for which elements will be eliminated. Yield surface can be scaled for strain rate dependence by the Cowper-Symonds model.

This material behavior is written as *MAT_ PIECEWISE_LINEAR_PLASTICITY (p. 181) , where the parameter lcss is the curve id of the effective stress versus plastic strain.

Name	Symbol	Units	Notes
Initial Yield Stress	A	Stress	

Name	Symbol	Units	Notes
Strain Rate Constant	C	None	Assumed 1/second in all cases
Strain Rate Constant	P	None	
Strain Rate Correction	-	None	Option List: Scale Yield Stress Viscoplastic

6.6.8. Modified Cowper-Symonds Piecewise Linear Hardening

This model is an enhanced version of the Cowper Symonds Piecewise Linear model that accounts for multiple failure methods:

- Effective plastic strain
- Thinning (through-thickness) plastic strain
- Major principal in-plane strain

The plastic strain failure parameter of this material model is defined by adding a plastic strain failure behavior to it.

Name	Symbol	Units	Notes
Initial Yield Stress	A	Stress	
Strain Rate Constant	C	None	Assumed 1/second in all cases
Strain Rate Constant	P	None	
Thinning Strain At Failure		None	
Major In Plane Strain At Failure		None	
Strain Rate Correction	-	None	Option List: Scale Yield Stress Viscoplastic

This material behavior is written as *MAT_ MODIFIED_PIECEWISE_LINEAR_PLASTICITY (p. 175) , where the parameter lcss is the curve id of the effective stress versus plastic strain.

6.7. Forming Plasticity

The following Forming Plasticity models are discussed in this section:

6.7.1. Bilinear Transversely Anisotropic Hardening

This material model is most commonly used for sheet metal forming of anisotropic materials.

It is a fully iterative anisotropic plasticity model available for shell and 2–D elements only. In this model the yield function given by Hill[3] is reduced to the following for the case of plane stress:

$$\sigma_y=\sqrt{\sigma_{11}^2+\sigma_{22}^2-\frac{2R}{R+1}\sigma_{11}\sigma_{22}+2\frac{2R+1}{R+1}\sigma_{12}^2}$$

The anisotropic hardening parameter, R, is defined by the ratio of the in-plane plastic strain rate to the out-of-plane plastic strain rate:

$$R=\frac{\dot{\varepsilon}_{22}^p}{\dot{\varepsilon}_{33}^p}$$

Name	Symbol	Units	Notes
Yield Strength		Stress	
Tangent Modulus		Stress	
Anisotropic hardening Parameter		None	

This material behavior is written as *MAT_TRANSVERSELY_ANISOTROPIC_ELASTIC_PLASTIC (p. 185) .

6.7.2. Multilinear Transversely Anisotropic Hardening

This material model is most commonly used for sheet metal forming of anisotropic materials.

It is a fully iterative anisotropic plasticity model available for shell and 2-D elements only. In this model the yield function given by Hill[3] is reduced to the following for the case of plane stress:

$$\sigma_y=\sqrt{\sigma_{11}^2+\sigma_{22}^2-\frac{2R}{R+1}\sigma_{11}\sigma_{22}+2\frac{2R+1}{R+1}\sigma_{12}^2}$$

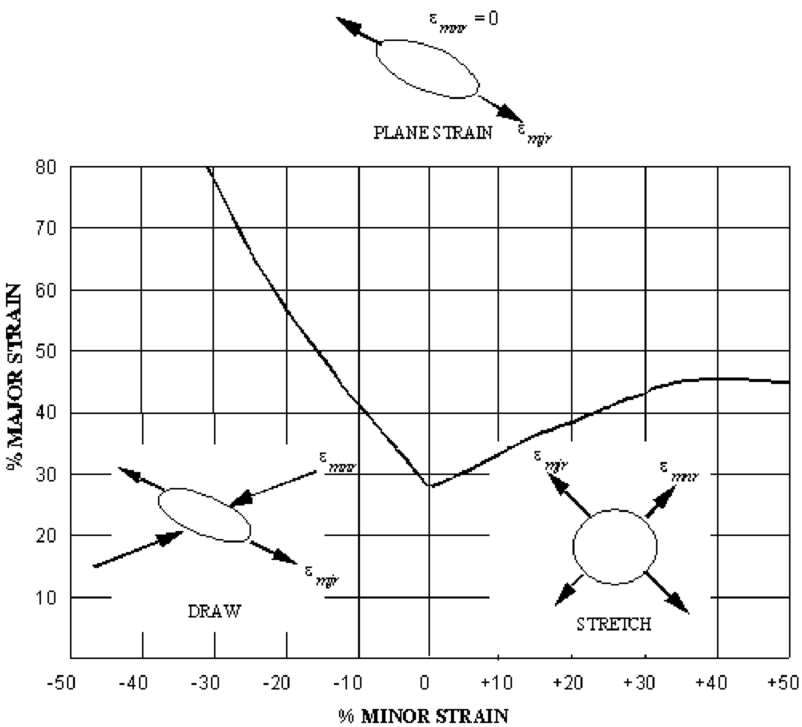
A load curve parameter is defined for the relationship between the effective yield stress and the effective plastic strain.

Name	Symbol	Units	Notes
Anisotropic hardening Parameter		None	

This material behavior is written as *MAT_TRANSVERSELY_ANISOTROPIC_ELASTIC_PLASTIC (p. 185) .

6.7.3. Bilinear FLD Transversely Anisotropic Hardening

This material model is used for simulating the sheet metal forming of anisotropic materials. Only transversely anisotropic materials can be considered. For this model, the dependence of the flow stress with the effective plastic strain is modeled by defining a yield stress and a tangent modulus. In addition, you also define a forming limit diagram. This diagram will be used to compute the maximum strain ratio that the material can experience.



This plasticity model is only available for shell and 2-D elements. The model directly follows the plasticity theory introduced in the Transversely Anisotropic Elastic Plastic model described earlier in this section. You can refer to that model for the theoretical basis.

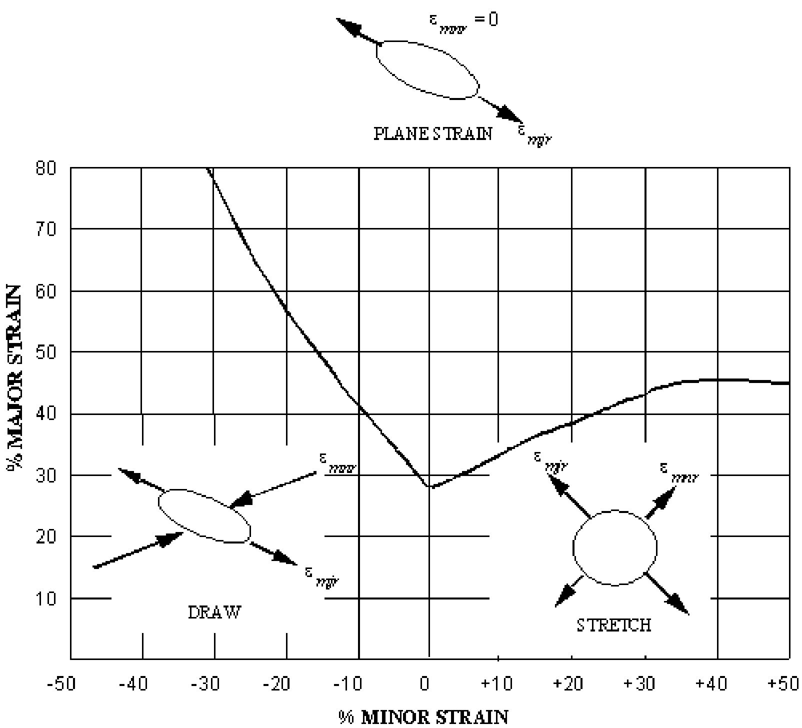
Name	Symbol	Units	Notes
Yield Strength		Stress	
Tangent Modulus		Stress	
Anisotropic hardening Parameter		None	

This material behavior is written as *MAT_FLD_TRANSVERSELY_ANISOTROPIC (p. 172) .

6.7.4. Multilinear FLD Transversely Anisotropic Hardening

This material model is used for simulating the sheet metal forming of anisotropic materials. Only transversely anisotropic materials can be considered. For this model, the dependence of the flow stress with the effective plastic strain is modeled using a curve. In addition, you also define a forming limit diagram. This diagram will be used to compute the maximum strain ratio that the material can experience.

This plasticity model is only available for shell and 2-D elements. The model directly follows the plasticity theory introduced in the Transversely Anisotropic Elastic Plastic model described earlier in this section. You can refer to that model for the theoretical basis.



Name	Symbol	Units	Notes
Anisotropic hardening Parameter		None	

This material behavior is written as *MAT_FLD_TRANSVERSELY_ANISOTROPIC (p. 172) .

6.7.5. Bilinear 3 Parameter Barlat Hardening

This is an anisotropic plasticity model developed by Barlat and Lian[1] used for modeling aluminum sheets under plane stress conditions. Both exponential and linear hardening rules are available. The anisotropic yield criterion for plane stress is defined as:

$$2(\sigma_Y)^m = a\left|K_1+K_2\right|^m + a\left|K_1-K_2\right|^m + c\left|2K_2\right|^m$$

where σ_Y is the yield stress, a and c are anisotropic material constants, m is Barlat exponent, and K_1 and K_2 are defined by:

$$K_1=\frac{\sigma_{xx}+h\sigma_{yy}}{2}$$

$$K_2=\sqrt{\left(\frac{\sigma_{xx}-h\sigma_{yy}}{2}\right)^2+p^2\tau_{xy}^2}$$

Here, h and p are additional anisotropic material constants. For the exponential hardening option, the material yield strength is given by:

$$\sigma_y = k\,(\varepsilon_0 + \varepsilon_p)\,n$$

where k is the strength coefficient, ε_0 is the initial strain at yield, ε_p is the plastic strain, and n is the hardening coefficient. All of the anisotropic material constants, excluding p which is determined implicitly, are determined from Barlat and Lian width to thickness strain ratio (R) values as shown:

$$a = 2 - 2\sqrt{\frac{R_{00}}{1+R_{00}}\,\frac{R_{90}}{1+R_{90}}}$$

c = 2 − a

$$h = \sqrt{\frac{R_{00}}{1+R_{00}}\,\frac{1+R_{90}}{R_{90}}}$$

The width to thickness strain ratio for any angle Φ can be calculated from:

$$R_\phi = \frac{2m\sigma_Y^m}{\left(\frac{\partial\Phi}{\partial\sigma_{xx}} + \frac{\partial\Phi}{\partial\sigma_{yy}}\right)\sigma_\phi} - 1$$

The hardening rule is linear and requires input of yield strength and tangent modulus, in addition to the Barlat exponent.

This material behavior is written as *MAT_3-PARAMETER_BARLAT (p. 167) .

6.7.6. Exponential 3 Parameter Barlat Hardening

The theory is identical to the bilinear 3 parameter Barlat model, but for this material model the hardening rule is exponential. It requires input of the hardening constant K, and hardening exponent, in addition to the Barlat exponent.

This material behavior is written as *MAT_3-PARAMETER_BARLAT (p. 167) .

6.7.7. Exponential Barlat Anisotropic Hardening

This is an anisotropic plasticity model developed by Barlat, Lege, and Brem[2] used for modeling material behavior in forming processes. The anisotropic yield function Φ is defined as:

$$\Phi = |S_1 - S_2|^m + |S_2 - S_3|^m + |S_3 - S_1|^m$$

where m is the flow potential exponent and S_i are the principal values of the symmetric matrix S_{ij} .

$$S_{xx} = 1/3\,[c\,(\sigma_{xx}-\sigma_{yy})-b\,(\sigma_{zz}-\sigma_{xx})]$$
$$S_{yy} = 1/3\,[a\,(\sigma_{yy}-\sigma_{zz})-c\,(\sigma_{xx}-\sigma_{yy})]$$
$$S_{zz} = 1/3\,[b\,(\sigma_{zz}-\sigma_{xx})-a\,(\sigma_{yy}-\sigma_{zz})]$$
$$S_{yz} = f\,\sigma_{yz}$$
$$S_{zx} = g\,\sigma_{zx}$$
$$S_{xy} = f\,\sigma_{xy}$$

where a, b, c, f, g, and h represent the anisotropic material constants. When a=b=c=f=g=h=1, isotropic material behavior is modeled and the yield surface reduces to the Tresca surface for m = 1 and the von Mises surface for m = 2 or 4. For this material option, the yield strength is given by:

$$\sigma_y = k\,(\varepsilon^P+\varepsilon_0)^n$$

where k is the strength coefficient, ε^P is the plastic strain, ε_0 is the initial strain at yield, and n is the hardening coefficient. The stress-strain behavior can be specified at only one temperature.

This material behavior is written as *MAT_BARLAT_ANISOTROPIC_PLASTICITY (p. 168) .

6.8. Foams

6.8.1. Rate Independent Low Density Foam

This is a highly compressible (urethane) foam material model often used for padded materials such as seat cushions. In compression, the model assumes hysteresis unloading behavior with possible energy dissipation. In tension, the material model behaves linearly until tearing occurs. For uniaxial loading, the model assumes that there is no coupling in transverse directions. By using input shape factor controls (a hysteresis unloading factor (HU), a decay constant (β), and a shape factor for unloading), the observed unloading behavior of foams can be closely approximated. The stress-strain behavior can be specified at only one temperature.

Input the curve for nominal stress vs. strain, the tension cutoff (tearing) stress, the hysteresis unloading factor, the decay constant, the viscous coefficient, and the shape factor for unloading,

6.9. Eulerian

6.9.1. Vacuum

This model is a dummy material representing a vacuum in a multi-material Euler/ALE model. Instead of using ELFORM = 12 (under *SECTION_SOLID), you should use ELFORM = 11 with the void material defined as vacuum material instead.

6.10. Rigid Materials

See the description in Rigid Materials .

6.11. Equations of State

Background information is discussed in this section along with available EOS models:

6.11.1. Background

A general material model requires equations that relate stress to deformation and internal energy (or temperature). In most cases, the stress tensor may be separated into a uniform hydrostatic pressure (all three normal stresses equal) and a stress deviatoric tensor associated with the resistance of the material to shear distortion.

Then the relation between the hydrostatic pressure, the local density (or specific volume) and local specific energy (or temperature) is known as an equation of state.

Hooke's law is the simplest form of an equation of state and is implicitly assumed when you use linear elastic material properties. Hooke's law is energy independent and is only valid if the material being modeled undergoes relatively small changes in volume (less than approximately 2%). One of the alternative equation of state properties should be used if the material is expected to experience high volume changes during an analysis.

Before looking at the various equations of state available, it is good to understand some of the fundamental physics behind their formulations. See the links in the following sections.

6.11.2. Ideal Gas EOS

One of the simplest forms of equation of state is that for an ideal polytropic gas which may be used in many applications involving the motion of gases. This material property is used by the Explicit Materials library in Engineering Data. See the description in Ideal Gas EOS for additional information.

This property requires that you also define Density and Specific Heat Constant Volume, Cv.

Written as *EOS_IDEAL_GAS (p. 164) and *MAT_NULL (p. 177).

6.11.3. Bulk Modulus

See the description in Bulk Modulus.

6.11.4. Shear Modulus

See the description in Shear Modulus.

6.11.5. Polynomial EOS

See the description in Polynomial EOS.

Written as *EOS_LINEAR_POLYNOMIAL (p. 164).

See Keywords used by LS-DYNA in Workbench (p. 129) for more information.

6.11.6. Shock EOS Linear

See the description in Shock EOS Linear.

Written as *EOS_GRUNEISEN (p. 163).

See Keywords used by LS-DYNA in Workbench (p. 129) for more information.

6.11.7. Shock EOS Bilinear

See the description in Shock EOS Bilinear.

Written as *EOS_GRUNEISEN (p. 163).

See Keywords used by LS-DYNA in Workbench (p. 129) for more information.

6.11.8. Explosive JWL

The JWL equation of state describes the detonation product expansion. This material property is used by the Explicit Materials library in Engineering Data. See the description in JWL EOS for additional information.

This property requires that you also define Density.

Written as *EOS_JWL (p. 165) and *MAT_HIGH_EXPLOSIVE_BURN (p. 171).

6.12. Failure

Background

Materials are not able to withstand tensile stresses which exceed the material's local tensile strength. The computation of the dynamic motion of materials assuming that they always remain continuous, even if the predicted local stresses reach very large values, will lead to unphysical solutions.

A model has to be constructed to recognize when tensile limits are reached to modify the computation to deal with this and to describe the properties of the material after this formulation has been applied.

Several different modes of failure initiation can be represented in the explicit dynamics system.

Element failure in the explicit dynamics system has two components, failure initiation and post failure response.

Failure Initiation

A number of mechanisms are available to initiate failure in a material (see properties Plastic Strain Failure, Principal Stress Failure, Principal Strain Failure, Tensile Pressure Failure, Johnson-Cook Failure). When specified criteria are met within an element, a post failure response is activated.

Post Failure Response

After failure initiation in an element, the subsequent strength characteristics of the element will change depending on the type of failure model.

- Instantaneous Failure

  Upon failure initiation, the element deviatoric stress will be immediately set to zero and retained at this level. Subsequently, the element will only be able to support compressive pressures.

By default, tensile failure models will produce an instantaneous post failure response.

The following Failure models are discussed in this section:

6.12.1. Plastic Strain Failure

Plastic strain failure can be used to model ductile failure in materials. Failure initiation is based on the effective plastic strain in the material. The user inputs a maximum plastic strain value.

If the material effective plastic strain is greater than the user defined maximum, failure initiation occurs. The material instantaneously fails.

Note:

This failure model must be used in conjunction with a plasticity or brittle strength model.

6.12.2. Principal Stress Failure

Principal stress failure can be used to represent brittle failure in materials.

Failure initiation is based on one of two criteria

- Maximum principal tensile stress
- Maximum shear stress (derived from the maximum difference in the principal stresses)

Failure is initiated when either of the above criteria is met. The material instantaneously fails.

If this model is used in conjunction with a plasticity model, it is often recommended to deactivate the Maximum Shear stress criteria by specifying a large value. In this case the shear response will be handled by the plasticity model.

Name	Symbol	Units	Notes
Maximum Tensile Stress		Stress	User must input a positive value. Default = +1e+20
Maximum Shear Stress		Stress	User must input a positive value. Default = +1e+20

6.12.3. Principal Strain Failure

Principal strain failure can be used to represent brittle or ductile failure in materials.

Failure initiation is based on one of two criteria

- Maximum principal tensile strain
- Maximum shear strain (derived from the maximum difference in the principal strains)

Failure is initiated when either of the above criteria is met. The material instantaneously fails.

If this model is used in conjunction with a plasticity model, it is often recommended to deactivate the maximum shear strain criteria by specifying a large value. In this case the shear response will be treated by the plasticity model.

Name	Symbol	Units	Notes
Maximum Principal Strain		None	User must input a positive value. Default = +1e+20
Maximum Shear Strain		None	User must input a positive value. Default = +1e+20

6.12.4. Johnson-Cook Failure

The Johnson-Cook failure model can be used to model ductile failure of materials experiencing large pressures, strain rates and temperatures.

$$D = \sum \frac{\Delta\varepsilon}{\varepsilon^f}$$

$$\varepsilon^f = \underbrace{\left[D_1 + D_2 e^{D_3 \sigma^*}\right]}_{\text{Pressure dependence}} \underbrace{\left[1 + D_4 \ln\left|\dot{\varepsilon}^*\right|\right]}_{\text{Strain rate dependence}} \underbrace{\left[1 + D_5 T^*\right]}_{\text{Temperature dependence}}$$

This model is constructed in a similar way to the Johnson-Cook plasticity model in that it consists of three independent terms that define the dynamic fracture strain as a function of pressure, strain rate and temperature:

Name	Symbol	Units	Notes
Damage Constant D1	D1	None	

Name	Symbol	Units	Notes
Damage Constant D2	D2	None	
Damage Constant D3	D3	None	
Damage Constant D4	D4	None	
Damage Constant D5	D5	None	
Melting Temperature		Temperature	

6.13. Thermal Properties

Properties
Isotropic Thermal Conductivity
Specific Heat

These properties are used when the LS-DYNA calculation is thermal-structural coupled, and are part of the definition of the *MAT_THERMAL_ISOTROPIC material.

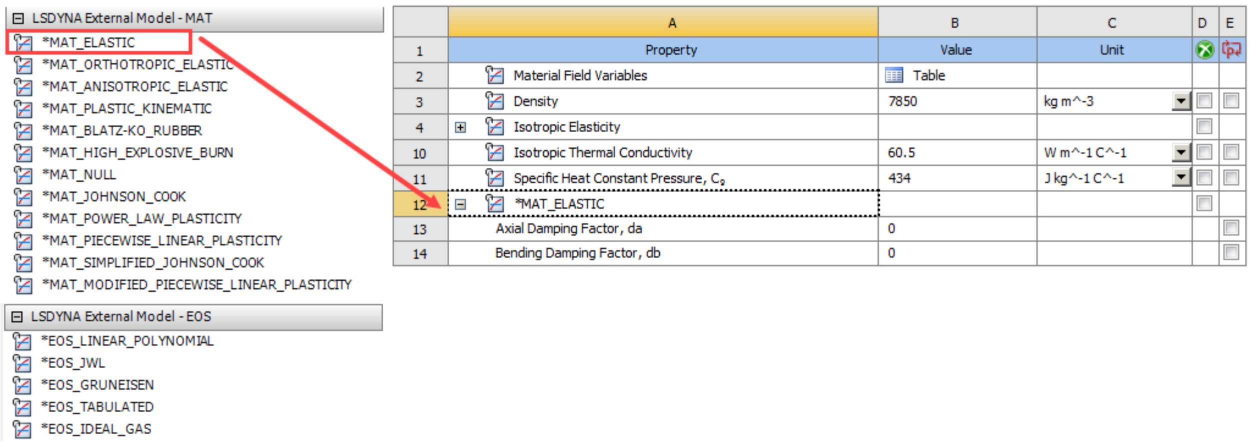
6.14. Electromagnetic Properties

See Electromagnetic Material Properties in the *Engineering Data User's Guide* for additional information about Electromagnetic material properties in Mechanical.

Properties	Notes
Isotropic Resistivity	This property is used when LS-DYNA calculation is electromagnetic-thermal-structural coupled and is part of the definition of the *EM_MAT_001 and *EM_MAT_002 cards.
Orthotropic Resistivity	This property is used when LS-DYNA calculation is electromagnetic-thermal-structural coupled and is part of the definition of the *EM_MAT_003 card.
Isotropic Relative Permeability	This property is used when the LS-DYNA calculation is electromagnetic-thermal-structural coupled and is part of the definition of the *EM_MAT_002 card. (Linear "Soft" Magnetic Material)

6.15. LS-DYNA External Model Material Properties

These properties are created when an LS-DYNA input file is imported using External Model. You can also add these properties directly to a material in Engineering Data by selecting them from the LSDYNA External Model - MAT or LSDYNA External Model - EOS sections of the Toolbox as shown in the following figure.



These properties are direct representations of the existing LS-DYNA material models, and enable additional fields for definition. Additional information can be found in LS-DYNA Keyword User's Manual Volume II , which describes these material models in detail.

Note:

When the optional Viscoelastic cards are imported into Engineering Data from External Model a conversion is made to the Prony series terms. The Shear\Bulk Moduli are converted to Relative Moduli by applying the Initial Shear\Bulk Modulus as a scale factor, and the Damping Coefficients are converted to Relaxation Times by inversion.

$\alpha_i^G = G_i / G_0$ and $\tau_i^G = 1/\beta_i^G$	$\alpha_i^K = K_i / K_0$ and $\tau_i^K = 1/\beta_i^K$
Where:	Where:
G_i = Shear Moduli	K_i = Bulk Moduli
α_i^G = Relative Moduli	α_i^K = Relative Moduli
G_0 = Initial Shear Modulus	K_0 = Initial Bulk Modulus
τ_i^G = Relaxation Times	τ_i^K = Relaxation Times
β_i^G = Damping Coefficients	β_i^K = Damping Coefficients

When writing the solver input file from the LS-DYNA analysis system the conversion is made from Relative Moduli to Shear\Bulk Moduli and Relaxation Times to Damping Coefficients.

$$G_i = \alpha_i^G G_0 \text{ and } \beta_i^G = 1/\tau_i^G$$

$$K_i = \alpha_i^K K_0 , \beta_i^K = 1/\tau_i^K$$

The Initial Shear Modulus for the Ogden Rubber Material Parameters formulation model is calculated using the definition in Ogden Hyperelasticity (**TB**,HYPER,,,,OGDEN) in the *Material Reference*. The other material models have a specific field in Engineering Data to define the Initial Shear\Bulk Modulus.

The Ogden Rubber Least Squares formulation model does not currently support Prony series.

6.15.1. *MAT_ELASTIC

This property requires that you also define Isotropic Elastic Elasticity and Density. It enables the definition of the Axial Damping Factor and Bending Damping Factor for Belytschko Schwer beams.

This material behavior is written as *MAT_ELASTIC (or *MAT_001) (p. 170) .

6.15.2. *MAT_ORTHOTROPIC_ELASTIC

This material behavior is written as *MAT_ORTHOTROPIC_ELASTIC (or *MAT_002) (p. 178) .

6.15.3. *MAT_ANISOTROPIC_ELASTIC

This material behavior is written as *MAT_ANISOTROPIC_ELASTIC (or *MAT_002_ANIS) (p. 179) .

6.15.4. *MAT_PLASTIC_KINEMATIC

This property requires that you also define Isotropic Elasticity.

This material behavior is written as *MAT_PLASTIC_KINEMATIC (or *MAT_003) (p. 181) .

6.15.5. *MAT_BLATZ-KO_RUBBER

This material behavior is written as *MAT_BLATZ-KO_RUBBER (or *MAT_007) (p. 170) .

6.15.6. *MAT_JOHNSON_COOK

This material behavior is written as *MAT_JOHNSON_COOK (or *MAT_015) (p. 174) .

6.15.7. *MAT_POWER_LAW_PLASTICITY

This property requires that you also define Isotropic Elastic Elasticity and Density.

This material behavior is written as *MAT_POWER_LAW_PLASTICITY (or *MAT_018) (p. 182) .

6.15.8. *MAT_PIECEWISE_LINEAR_PLASTICITY

This property requires that you also define Isotropic Elastic and Density.

This material behavior is written as *MAT_PIECEWISE_LINEAR_PLASTICITY (or *MAT_24) (p. 181) .

6.15.9. *MAT_SIMPLIFIED_JOHNSON_COOK

This property requires that you also define Isotropic Elastic Elasticity and Density.

This material behavior is written as *MAT_SIMPLIFIED_JOHNSON_COOK (or *MAT_098) (p. 184) .

6.15.10. *MAT_MODIFIED_PIECEWISE_LINEAR_PLASTICITY

This property requires that you also define Isotropic Elastic Elasticity and Density.

This material behavior is written as *MAT_MODIFIED_PIECEWISE_LINEAR_PLASTICITY (or *MAT_123) (p. 175) .

6.15.11. *MAT_HIGH_EXPLOSIVE_BURN

This property requires that you also define Density.

This material behavior is written as *MAT_HIGH_EXPLOSIVE_BURN (*MAT_008).

6.15.12. *MAT_NULL

This property requires that you also define Density.

This material behavior is written as *MAT_NULL (*MAT_009).

6.15.13. *EOS_LINEAR_POLYNOMIAL

This material behavior is written as *EOS_LINEAR_POLYNOMIAL (*EOS_001) (p. 164) .

6.15.14. *EOS_JWL

This material behavior is written as *EOS_JWL (*EOS_002).

6.15.15. *EOS_GRUNEISEN

This material behavior is written as *EOS_GRUNEISEN (*EOS_004) (p. 163) .

6.15.16. *EOS_TABULATED

This material behavior is written as *EOS_TABULATED (*EOS_009).

6.15.17. *EOS_IDEAL_GAS

This material behavior is written as *EOS_IDEAL_GAS (*EOS_012).

6.15.18. *EOS_MURNAGHAN

This material behavior is written as *EOS_MURNAGHAN (*EOS_019). This EOS was designed to model incompressible fluid flow with SPH or ALE elements.

6.15.19. *EM_EOS_TABULATED1

This material behavior is written as *EM_EOS_TABULATED1.

This property requires that you also define an EM_MAT. This property is not compatible with EM_MAT_003, EM_MAT_005, and EM_MAT_006. From Engineering Data you can only define an equation of state for conductivity.

6.15.20. *EM_MAT_001

This material behavior is written as *EM_MAT_001.

This property requires that you also define Elasticity, Density, Thermal Conductivity, and Heat Capacity.

6.15.21. *EM_MAT_002

This material behavior is written as *EM_MAT_002.

This property requires that you also define Elasticity, Density, Thermal Conductivity, and Heat Capacity.

6.15.22. *EM_MAT_003

This material behavior is written as *EM_MAT_003.

This property requires that you also define Elasticity, Density, Thermal Conductivity, and Heat Capacity.

6.15.23. *EM_MAT_004

This material behavior is written as *EM_MAT_004.

This property requires that you also define Elasticity, Density, Thermal Conductivity, and Heat Capacity.

6.15.24. *EM_MAT_005

This material behavior is written as *EM_MAT_005.

This property requires that you also define Elasticity, Density, Thermal Conductivity, and Heat Capacity.

6.15.25. *EM_MAT_006

This material behavior is written as *EM_MAT_006.

This property requires that you also define Elasticity, Density, Thermal Conductivity, and Heat Capacity.

6.15.26. *MAT_CRUSHABLE_FOAM

This property requires that you also define Isotropic Elastic Elasticity and Density.

This material behavior is written as *MAT_CRUSHABLE_FOAM (*MAT_063).

6.15.27. *MAT_SIMPLIFIED_RUBBER/FOAM

This property requires that you also define Density. The optional viscoelastic constants cards will be written if the constants have been defined using the Prony Shear Relaxation properties in Engineering Data.

This material behavior is written as *MAT_SIMPLIFIED_RUBBER/FOAM (*MAT_181).

6.15.28. *MAT_OGDEN_RUBBER

This property requires that you also define Density. The optional viscoelastic constants cards will be written if the constants have been defined using the Prony Shear Relaxation properties in Engineering Data.

This material behavior is written as *MAT_OGDEN_RUBBER (*MAT_077_O).

6.15.29. *MAT_GENERAL_VISCOELASTIC

This property requires that you also define Density and Bulk Modulus. The optional viscoelastic constants cards will be written if the constants have been defined using the Prony Shear Relaxation and Prony Volumetric Relaxation properties in Engineering Data. The optional Shift Function fields can be defined using the Williams-Landel-Ferry Shift Function properties in Engineering Data.

This material behavior is written as *MAT_GENERAL_VISCOELASTIC (*MAT_076).

6.15.30. *MAT_ BILKHU/DUBOIS_FOAM

This property requires that you also define Isotropic Elastic Elasticity and Density.

This material behavior is written as *MAT_BILKHU/DUBOIS_FOAM (*MAT_075).

6.15.31. *MAT_FABRIC

This property requires that you also define Density.

This material behavior is written as *MAT_FABRIC (*MAT_034).

6.15.32. *MAT_COMPOSITE_FAILURE_SHELL_MODEL

This material behavior is written as *MAT_COMPOSITE_FAILURE_SHELL_MODEL.

6.15.33. *MAT_COMPOSITE_FAILURE_SOLID_MODEL

This material behavior is written as *MAT_COMPOSITE_FAILURE_SOLID_MODEL.

6.15.34. *MAT_ENHANCED_COMPOSITE_DAMAGE

This material behavior is written as *MAT_ENHANCED_COMPOSITE_DAMAGE.

6.15.35. *MAT_LAMINATED_COMPOSITE_FABRIC

This material behavior is written as *MAT_LAMINATE_COMPOSITE_FABRIC.

6.15.36. *MAT_ORTHOTROPIC _SIMPLIFIED_DAMAGE

This material behavior is written as *MAT_ORTHOTROPIC _SIMPLIFIED_DAMAGE.

6.15.37. *MAT_ADD_DAMAGE_GISSMO

This material behavior is written as *MAT_ADD_DAMAGE_GISSMO.

To define a valid material in workbench, the following conditions must be true:

- ECRIT must be greater than or equal to 0
- FADEXP must be greater than or equal to 0
- LCSRS must be a curve (table is not supported)

6.15.38. *MAT_ADD_EROSION

This material behavior is written as *MAT_ADD_EROSION.

To define a valid material in workbench, the following conditions must be true:

- SIGVM must be greater than or equal to 0
- MXEPS must be greater than or equal to 0
- LCFLD must be a curve (table is not supported)
- IDAM must be equal to 0

6.15.39. *MAT_ALE_VISCOUS

This material behavior is written as *MAT_ALE_VISCOUS.

Chapter 7: Customizing LS-DYNA using ACT

LS-DYNA can be customized with ACT (Ansys Customization Toolkit). You can create interactions between user-defined extensions and the LS-DYNA system. You can create objects or features for custom preprocessing in the context of Ansys Mechanical. Custom postprocessing is not supported.

This section assumes some degree of familiarity with ACT. Other Ansys guides provide related information:

- For an introduction to writing scripts for Mechanical, see the *Scripting in Mechanical Guide*.
- For descriptions of all ACT API objects, methods, and properties, see the *ACT API Reference Guide*.
- For information on how to use ACT to create apps (extensions) that customize and automate Ansys products, see the *ACT Developer's Guide*.
- For ACT usage, customization, and automation information specific to Mechanical, see the *ACT Customization Guide for Mechanical*.

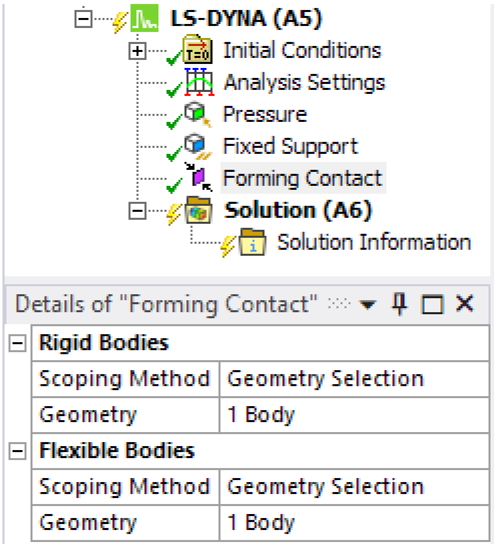
An example of how to set up custom preprocessing is described here.

The following is an example from an XML file that sets up Rigid Bodies and Flexible Bodies categories in the Data panel, each with a Geometry field. The resulting load and Details panel are shown.

```
<simdata context="Mechanical">
    <load name="Forming" version="1" caption="Forming Contact" icon="contact" issupport="false" isload="true" color
      <attributes custom_dyna="Yes"/>
      <propertygroup name="Rigid Bodies"  caption="Rigid Bodies">
        <property name="Geometry" control="scoping">
          <attributes>
            <selection_filter>body</selection_filter>
          </attributes>
        </property>
      </propertygroup>

      <propertygroup name="Flexible Bodies"  caption="Flexible Bodies">
        <property name="Geometry" control="scoping">
          <attributes>
            <selection_filter>body</selection_filter>
          </attributes>
        </property>
      </propertygroup>
      <callbacks>
        <getcommands location="pre" >writeSnippet</getcommands>
      </callbacks>

    </load>
  </simdata>
```



To enable a custom load for LS-DYNA usage, define the attribute `custom_dyna` for the load, and set its value to **yes**. Additionally, if the load needs to write information to the solver input file, the callback `getcommands` must be implemented. The `getcommands` element specifies the name of the IronPython function called when Mechanical writes the solver input file.

```
<getcommands location = [string] order = [integer]>-->
[function(solver,solverdata,stream)]</getcommands>
```

The `solverdata` is a helper structure. It helps integrate user extensions with LS-DYNA, particularly when the solver commands generated by the extension add entities that have identifiers, like nodes, elements or components. Through the `solverdata` structure, ACT provides an efficient method to manage the newly created IDs of these entities with the existing IDs previously generated by LS-DYNA. It also enables reuse of internally implemented keywords like materials or elements.

Note:

The syntax shown for the methods is C#.

The following table and sections describe the available `solverdata` methods.

Table 7.1: Solverdata Methods

CreateMaterial (p. 255)CreateMaterial (p. 255)	Two different methods available to create materials for LS-DYNA.
CreateNewElement (p. 256)	Method to create an element keyword.
GetNewPartId (p. 257)	Method to get an unused part identifier.
LSDynaSolverExtension.KeyWords.Part.Part CreateNewPart (p. 257)	Method to create a new part keyword.
CreateSection (p. 257)	Method to create a new section keyword.
GetComponent (p. 258)	Method to get the LS-DYNA equivalent of a mechanical Named Selection.
GetContactId (p. 258)	Method to get the solver contact identifier of the contact object.

GetContactTargetId (p. 258)	Method to get the solver contact identifier of the contact target object.
GetCoordinateSystemSolverId (p. 259)	Method to get the solver coordinate system identifier.
GetEndTime (p. 259)	Method to get the simulation end time.
GetMaterialSolverId (p. 259)	Method to get the material identifier of a body.
GetNamedSelectionLSDYNAId (p. 260)	Method to get the solver named selection identifier.
GetNewContact (p. 260)	Method to get an empty contact keyword.
GetNewCurveId (p. 260)	Method to get an unused curve identifier.
GetNewElementId (p. 260)	Method to get an unused element identifier.
GetNewElementType (p. 260)	Method to get an unused part identifier.
GetNewNodeId (p. 261)	Method to get an unused node identifier.
GetNewVectorId (p. 261)	Method to get an unused vector identifier.
GetRemotePointNodeId (p. 261)	Method to get the master node identifier for a remote point.
GetSolverUnitSystem (p. 261)	Method to get the solver unit system.
ContainsDynamicRelaxation (p. 261)	Method to determine if dynamic relaxation is used in the model.
CurrentStep (p. 262)	Method to get the current step.
MaxElementId (p. 262)	Method to get the maximum element identifier.
MaxElementType (p. 262)	Method to get the maximum part identifier.
MaxNodeId (p. 262)	Method to get the maximum node identifier.

7.1. CreateMaterial

This method tries to create a LS-DYNA material keyword based on the Engineering Data material object. For example. If the material is an elastic only material, this method will return *MAT_ELASTIC keyword.

Declaration Syntax

public IMaterialKeyWord CreateMaterial(IMaterial materialClass, long materialId);

Table 7.2: Properties

Property	Type	Description
materialClass	IMaterial	The class of the Engineering Data material object.
materialID	long	The ID of the Engineering Data material object.

7.2. CreateMaterial

This method tries to create a LS-DYNA material keyword based on the Engineering Data material object and the material type. For example, if the material is compatible with an LS-DYNA spot-weld material (has at least elastic properties) and *MAT_SPOTWELD is entered as the material type, this method will return *MAT_SPOTWELD keyword.

Declaration Syntax

public IMaterialKeyWord CreateMaterial(IMaterial materialClass, string materialType, long materialId);

Table 7.3: Properties

Property	Type	Description
materialClass	IMaterial	The class of the Engineering Data material object.
materialType	string	The type of the Engineering Data material object.
materialID	long	The ID of the Engineering Data material object.

7.3. CreateNewElement

This method creates an element keyword (solid, shell, beam, etc.) based `elementType` and will return the LS-DYNA keyword that represents the specified element type.

Declaration Syntax

public IKeyWord CreateNewElement(string elementType);

The following element types are available:

Table 7.4: Properties

Property	Type	Description
elementType	string	The following values are valid: • "*ELEMENT_SOLID": • "*ELEMENT_SHELL": • "*ELEMENT_BEAM": • "*ELEMENT_DISCRETE": • "*ELEMENT_INERTIA": • "*ELEMENT_MASS": • "*ELEMENT_TSHELL":

7.4. GetNewPartId

This method returns an unused part identifier.

Declaration Syntax

public long GetNewPartId() ;

7.5. LSDynaSolverExtension.KeyWords.Part.Part CreateNewPart

This method will create a new part keyword with the identifier *`partId`*. By default, the material identifier and the section identifier are set to *`partId`*. This method should be used in conjunction with the method `GetNewPartId()` to avoid identifier clashes in the input file.

Declaration Syntax

public LSDynaSolverExtension.KeyWords.Part.Part CreateNewPart(long partId);

Table 7.5: Properties

Property	Type	Description
partId	long	The ID of the newly created part.

7.6. CreateSection

This method will create a new section keyword based on the section type *`sectionName`*, on the section identifier *`secId`*, and the element formulation *`elform`*.

Declaration Syntax

public IKeyWord CreateSection(string sectionName, long secId, long elform);

The following section types are available:

Table 7.6: Properties

Property	Type	Description
elementType	string	The following values are valid: • "*SECTION_SOLID" • "*SECTION_SECTION_ALE" • "*SECTION_SHELL" • "*SECTION_BEAM"

Property	Type	Description
		• "SECTION_DISCRETE"

7.7. GetComponent

This method returns a string containing the LS-DYNA text equivalent of a Mechanical Named Selection given an ACT *selectionInfo*. The identifier of the component identifier is automatically incremented by LS-DYNA.

Declaration Syntax

public string GetComponent(ISelectionInfo info, out int nsid);

Table 7.7: Properties

Property	Type	Description
info	ISelectionInfo	An ACT selection object.
nsid	int	A new ID for the named selection created.

7.8. GetContactId

This method returns the solver contact identifier given the tree identifier of the contact object.

Declaration Syntax

public string GetContactId(int treeId);

Table 7.8: Properties

Property	Type	Description
treeId	int	Workbench ID of contact object.

7.9. GetContactTargetId

This method returns the solver contact target identifier given the tree identifier of the contact object.

Declaration Syntax

public string GetContactTargetId(int treeId);

Table 7.9: Properties

Property	Type	Description
treeId	int	Workbench ID of contact target object.

7.10. GetCoordinateSystemSolverId

This method returns the solver coordinate system identifier given the tree identifier of the coordinate system object.

Declaration Syntax

public string GetCoordinateSystemSolverId(int id);

Table 7.10: Properties

Property	Type	Description
id	int	Workbench ID of coordinate system object.

7.11. GetEndTime

This method returns the simulation end time.

Declaration Syntax

public double GetEndTime();

7.12. GetMaterialSolverId

This method returns the material identifier given the body identifier the material is applied to.

Declaration Syntax

public string GetMaterialSolverId(int bodyId);

Table 7.11: Properties

Property	Type	Description
bodyId	int	Workbench ID of the body to which the material is applied.

7.13. GetNamedSelectionLSDYNAId

This method returns the LS-DYNA solver named selection identifier given an existing Workbench named selection.

Declaration Syntax

public int GetNamedSelectionLSDYNAId(ISelectionInfo namedSelection);

Table 7.12: Properties

Property	Type	Description
namedSelection	ISelectionInfo	Workbench ID of named selection of interest.

7.14. GetNewContact

This method returns a new empty contact keyword.

Declaration Syntax

public IKeyWord GetNewContact();

7.15. GetNewCurveId

This method returns an unused curve identifier.

Declaration Syntax

public ulong GetNewCurveId();

7.16. GetNewElementId

This method returns an unused element identifier.

Declaration Syntax

public ulong GetNewElementId();

7.17. GetNewElementType

This method returns an unused part identifier.

Declaration Syntax

public uint GetNewElementType();

7.18. GetNewNodeId

This method returns an unused node identifier.

Declaration Syntax

public ulong GetNewNodeId();

7.19. GetNewVectorId

This method returns an unused vector identifier.

Declaration Syntax

public ulong GetNewVectorId();

7.20. GetRemotePointNodeId

This method returns the master node identifier for a given remote point (tree) identifier.

Declaration Syntax

public int GetRemotePointNodeId(int remotePointId);

Table 7.13: Properties

Property	Type	Description
remotePointId	int	Workbench ID of remote point of interest.

7.21. GetSolverUnitSystem

This method returns the solver unit system.

Declaration Syntax

public string GetSolverUnitSystem() ;

7.22. ContainsDynamicRelaxation

This method returns whether dynamic relaxation is used or not in the model.

Declaration Syntax

public bool ContainsDynamicRelaxation { get; };

7.23. CurrentStep

This method returns the current step of the model. For LS-DYNA it is always 1.

Declaration Syntax

public uint CurrentStep { get; };

7.24. MaxElementId

This method returns the maximum element identifier in the model.

Declaration Syntax

public ulong MaxElementId { get; } ;

7.25. MaxElementType

This method returns the maximum part identifier in the model.

Declaration Syntax

public uint MaxElementType { get; };

7.26. MaxNodeId

This method returns the maximum node identifier in the model.

Declaration Syntax

public ulong MaxNodeId { get; };

Chapter 8: References

The following references are cited in this guide:

1. F. Barlat and J. Lian. "*Plastic Behavior and Stretchability of Sheet Metals*. Part I: A Yield Function for Orthotropic Sheets Under Plane Stress Conditions". *Int. Journal of Plasticity*, Vol. 5. pg. 51-66. 1989.

2. F. Barlat, D. J. Lege, and J. C. Brem. "*A Six-Component Yield Function for Anistropic Materials*". *Int. Journal of Plasticity*, Vol. 7. pg. 693-712. 1991.

3. R. Hill. "*A Theory of the Yielding and Plastic Flow of Anisotropic Metals*". *Proceedings of the Royal Society of London*, Series A., Vol. 193. pg. 281–197. 1948.