MpCCI Mapper

Version 4.7.0

MpCCI Mapper 4.7.0Documentation
PDF version
July 23, 2021

http://www.mpcci.de/en/mpcci-software/mapper.html

Fraunhofer
SCAI

Fraunhofer Institute for Algorithms and Scientific Computing SCAI
Schloss Birlinghoven, 53754 Sankt Augustin, Germany

Contents

1 Release Notes 5
1.1 Version 4.7.0 . . . 5
1.2 Version 4.6.0 . . . 6
1.3 Version 4.5.3 . . . 7
1.4 Version 4.5.2 . . . 7

2 Installation 8
2.1 Tool . . . 8
2.2 FlexLM . . . 8

3 Using the MpCCI Mapper 12
3.1 Menus . . . 12
3.2 Toolbar . . . 16
3.3 Viewport . . . 17
3.4 Data Panel . . . 18
3.5 Postprocessing . . . 26
3.6 Mapping - Theory of Algorithms in MpCCI Mapper . . . 28
3.7 Mapping with Mapper . . . 30
3.8 Batch Mode . . . 31

4 Solver Formats 36
4.1 Abaqus . . . 36
4.2 ANSYS Mechanical APDL . . . 38
4.3 AutoForm R4 converter . . . 40
4.4 Cadmould . . . 41
4.5 Indeed . . . 41
4.6 MSC Simufact . . . 42
4.7 Forge . . . 43
4.8 LSDyna . . . 44
4.9 Marc / COPRA FEA . . . 49
4.10 Moldex3D . . . 50
4.11 Moldflow . . . 51
4.12 Nastran . . . 51
4.13 OptiStruct . . . 52
4.14 PAM . . . 53
4.15 Radioss . . . 53
4.16 Sysweld . . . 54

5 Other Formats 55
5.1 Argus / Aramis . . . 55
5.2 Atos . . . 55
5.3 AutoGrid . . . 55
5.4 STL . . . 55
5.5 VTK . . . 56

6 Thirdparty License Information 57

6.1 Boost 57
6.2 LibQxt 57
6.3 Libtiff 58
6.4 Libstdc++ 58
6.5 Libwebp 58
6.6 LibXBC 59
6.7 pugixml 60
6.8 HDF5 60
6.9 PCRE Library 61
6.10 VTK 62

1 Release Notes

1.1 Version 4.7.0

List of changes and improvements since MpCCI Mapper version 4.6.0

1. Recent Files has been added to the File Menu 3.2 to quick access last five source and target files read by MpCCI Mapper
2. Settings Writer menu has been renamed to Interfaces to configure readers as well 3.6
3. Settings Viewer menu allows configuration of viewport background color and scalar bar color map 3.7
4. Transform Model feature was introduced to allow user defined translation or scaling of models
5. Reproduce Model feature was introduced to allow replication / cloning of meshes, e.g. sheet metal profiles or spot welds
6. Solid to shell thickness computation variant Experimental (8)
7. DISPLACEMENTS can be applied on source meshes after import and target meshes before export
8. Projection of target shell models onto source geometry shape 3.4.1

Table 1.1: GUI related changes and new features

8. Solver LSDyna 4.8
    8.1. ELEMENT_SHELL_COMPOSITE with material assignment 4.8.5
    8.2. MATFEM material orienation by invariant node numering
    8.3. NIPTETS can be edited in Overwrite Integration Rule dialog
    8.4. Maximal number of NIP Overwrite Integration Rule dialog
9. Solver Moldflow 4.11
    9.1. *.udm model file are supported for geometry
    9.2. Midplane meshes are now supported
    9.3. *.ele* files with fiber orientation can be imported for midplane meshes
    9.4. Part thickness can be read from *.udm file
10. New Solver format Moldex3D 4.10
11. New Solver format OptiStruct 4.13

Table 1.2: Solver formats changes and extension

1.2 Version 4.6.0

List of changes and improvements since MpCCI Mapper version 4.5.3

1. New mapping algortihmn for quantity ORIENTATION_TENSOR, based on Log-Euclidean framework, see 3.6.2 and 3.19 for detailed description
2. Quantity validation has been extended by a statistic report measuring minimal and maximal differences in data together with an arithmetic mean and standard deviation information 3.4.3
3. Quantity plot feature added to display the relative distribution of a quantity 3.5.3
4. Change display of quantity units in toolsbar quantity dialog and quantity range clip dialog 3.2
5. Transformation matrix storage option for LSDyna added; create INCLUDE_TRANFORM card for source or target model 3.4.1
6. Disable or enable mapping quantities in quantity context menu 3.4.2
7. Pyramid element with five nodes has been added to MpCCI Mapper

Table 1.3: GUI related changes and new features

4. Solvers MSC.Simufact Welding, Forming and Additive 4.6
   - 4.1. Universal file format redesigned as standalone interface
   - 4.2. Multiple bodies in single model now supported
   - 4.3. Support for layered solid elements
   - 4.4. Card 2414 format extended for data on integration points
   - 4.5. Automatic detection of STRESS, PLASTIC_STRAIN, TEMPERATURE and DISPLACEMENTS
5. Solver LS-Dyna 4.8
   - 5.1. Export of DISPLACEMENTS as deformed geometry and export nodal TEMPERATURE
   - 5.2. Enhanced support for MATFEM model material orientations
   - 5.3. Redefinition of element formulation (Elform) and through thickness integration scheme for target models 4.5
6. Solver Moldflow 4.11
   - 6.1. Geometry model file *.udm supported
   - 6.2. Import multiple *.xml on file open process
   - 6.3. Transient *.xml files supported; last defined state is used for import
   - 6.4. Optimized IO performance on XML data
7. Solver ANSYS Mechanical APDL 4.2
   - 7.1. List of supported solid elements has been extended
   - 7.2. INISTATE result format redesigned
   - 7.3. Export of DISPLACEMENTS as nblock with deformed nodal coordinates
8. Vendor-neutral standard for CAE data storage format VMAP added `https://www.vmap.eu.com`

Table 1.4: Solver formats changes and extension

1.3 Version 4.5.3

List of changes and improvements since MpCCI Mapper version 4.5.2

1. Added selection dialog if file extension matches more than one reader interface.
2. Model symmetry definition on source models to build symmetric part before mapping.
3. Vector data display has been revised. Data is automatically scaled to average element size. Only quantity DISPLACEMENTS is shows with actual length.
4. Element slice tool fixed for multiple part models.
5. QT library has been update to version 5.9.7.

Table 1.5: GUI related changes and new features

6. Radioss state file has been set as default export format.
7. LS-Dyna vairable TEMPERATURE support on export. INITIAL_STRAIN with variable number of points through shell thickness support 4.8.
8. Universal file format (Forge, Simufact) unit system detection. Fixed card 2414 format import on multiple data definition 4.7.
9. Abaqus fix on STRESS import.

Table 1.6: Solver formats changes and extension

1.4 Version 4.5.2

List of changes and improvements since MpCCI Mapper version 4.5.1

1. Graphical user interfaces has been redesigned and is now QT based 3
2. Validation concept has been redesigned and is now available for all quantities 3.4.3
3. Enhanced scriptable batch mode using configuration file 3.8.5
4. Enhanced data visualization for tensor and vector quantities 3.4.2

Table 1.7: GUI related changes and new features

5. Abaqus export of shell section composite when mapping fibre orientations 4.1
6. ANSYS export support for THICKNESS variable 4.2
7. COPRA FEA reading support (based on MSC Marc) 4.9
8. LSDyna history variable concept has been extended to assign higher history ids than defined
9. MSC Marc .t19 result file reading support 4.9
10. Moldflow can handle multiple XML-based result files at once 4.11
11. MSC Nastran SOL 700 write support for stresses and strains 4.12

Table 1.8: Solver formats changes and extension

2 Installation

2.1 Tool

MpCCI Mapper comes along as packed archive containing a platform specific executable 'mapper.exe' which can be unpacked to an arbitrary working directory and is ready to use after setting up necessary license information (c.f. section 2.2.2 for detailed description). On samba mounted directories take care of the execute flags to be set for the mapper.exe.

2.2 FlexLM

MpCCI Mapper uses a FlexLM based floating license mechanism. FlexLM license server has to be started on the license server host defined in the MpCCI Mapper license file. You can run MpCCI Mapper anywhere in your internal network. For more information about FlexLM, please refer to the FlexLM end-user's guide which is normally located in `<MAPPER_HOME>/license/LicenseAdministration.pdf`).

2.2.1 Prerequisits

In general FlexLM comes with some tools (lmutil, lmstat, lmhostid, lmgrd, lmdown etc.) for managing the licenses - please refer to the FlexLM documentation.

With MpCCI Mapper all FlexLM platform dependent binary executables are installed in a separate directory (`<MAPPER_HOME>/license/<architecture>`). Your license file should have the name 'mpcci_SVD_Your_Company_Expiration_Date.lic'.

MpCCI Mapper only needs three FlexLM executables:

the utility:	lmutil
the license server:	lmgrd
the vendor daemon:	SVD

The MpCCI Mapper vendor daemon is named SVD. The FlexLM port number of the SVD vendor daemon is 47000 by default. If there are other software packages installed using also FlexLM there will be several FlexLM utils available. Depending on your local installation and your own PATH environment it is not always defined which of these lmutils will be executed upon a command call.

2.2.2 Installing License

After installation of MpCCI Mapper you need to acquire a license file from Fraunhofer SCAI.

Please login on the host where the FlexLM license server for MpCCI Mapper should run on. In the following example, "$" is your prompt:

```
$ hostname
myHostName
```

```
$ lmutil lmhostid -n
12345abcd
```

Please send hostname (in the above example: "myHostName") and hostid to "mpcci@scai.fraunhofer.de".
If you have multiple network devices installed (Ethernet card, Wireless LAN, Docking Station on a Notebook) you may see multiple hostids. For the MpCCI Mapper license daemon the integrated ethernet card address should be the correct one.
Then, please start the FlexLM license server daemon and SVD vendor daemon on the local host with the following command:

```
$ lmgrd -c <LICENSE_FILE>
```

To stop the license server please type:

```
$ lmutil lmdown -c <LICENSE_FILE>
```

You should set the generic FlexLM variable SVD_LICENSE_FILE.
For ksh or bash user:

```
$ export SVD_LICENSE_FILE=47000@<HOSTNAME>
```

For csh users:

```
$ setenv SVD_LICENSE_FILE 47000@<HOSTNAME>
```

This variable is also referred to by other software packages using FlexLM;
LM_LICENSE_FILE may comprise several entries <port@host> (one for each application). If there is more than one entry they must be separated by a ":" on Unix/Linux and by a ";" on MS Windows.
We suggest that you set the SVD_LICENSE_FILE environment variable in your login file (.cshrc or .profile or .bashrc). FlexLM stores data of a previous successful connection to a license server in the file "<HOME>/.flexlmrc". You may create or adjust this file "<HOME>/.flexlmrc".
After setting this environment variable you can check whether the license server is running and can be reached from the machine where you are currently working and from where you plan to start MpCCI Mapper :

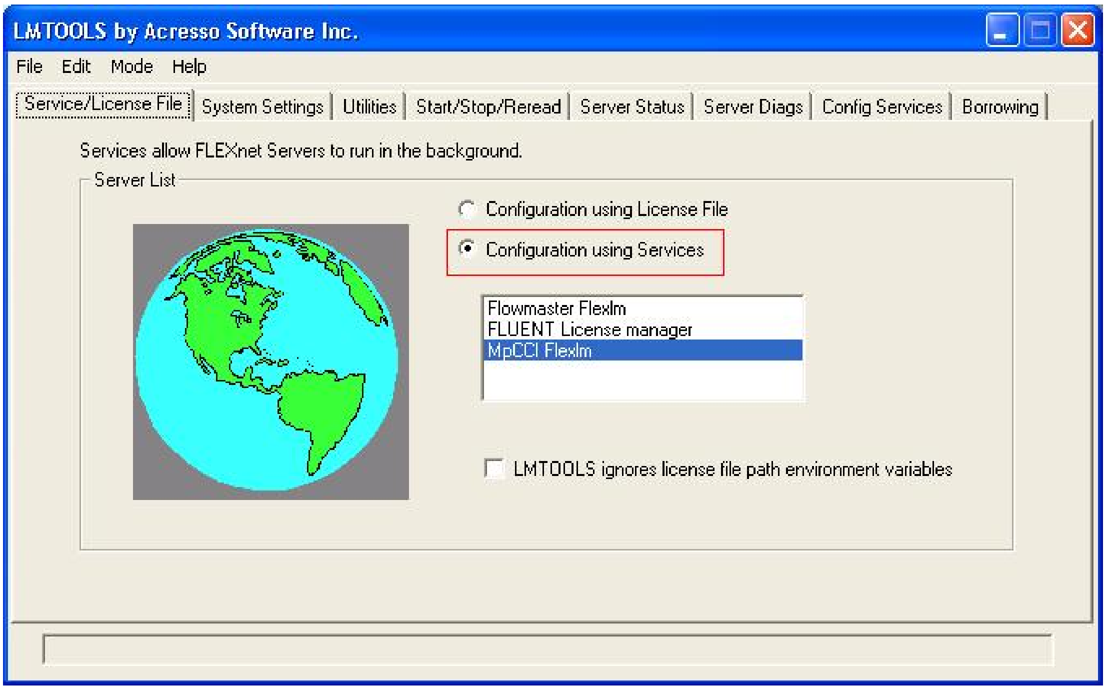
```
$ lmutil lmstat -vendor SVD
```

For more details

```
$ lmutil lmstat -a -c <LICENSE_FILE>
```

2.2.3 Configure a License Manager as Windows service

Execute the *lmtools.exe* application from the license manager installation directory:
$< LICENSE\_TOOL\_INSTALLATION > /bin/lmtools.exe$



Services allow FLEXnet Servers to run in the background.

Server List

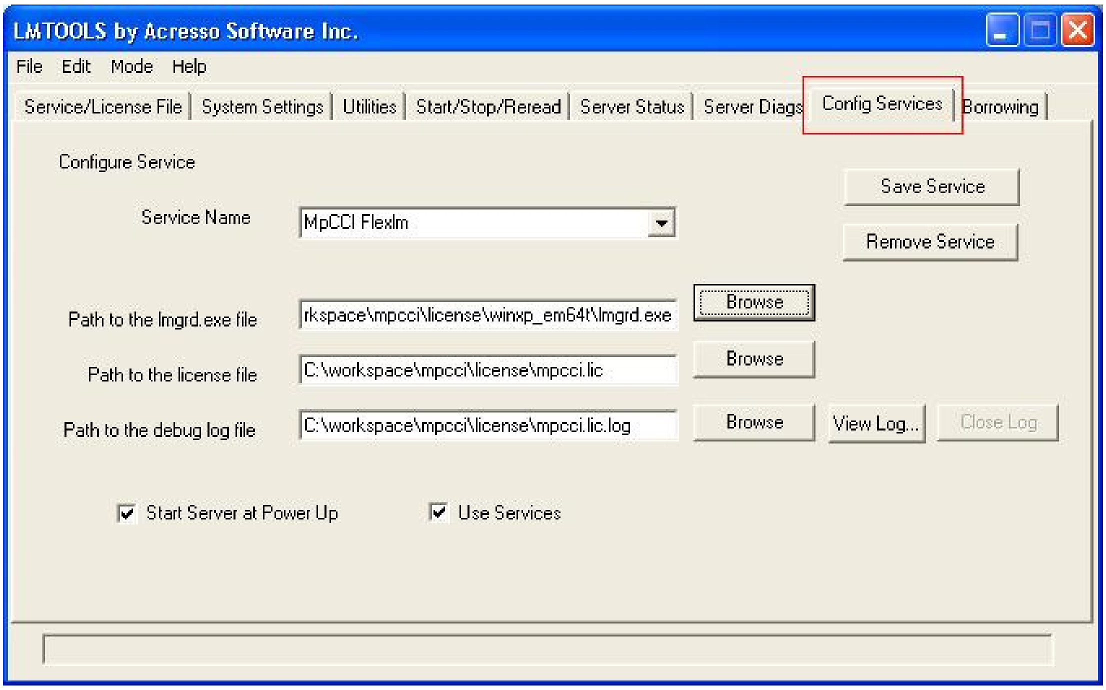
○ Configuration using License File

● Configuration using Services

Flowmaster Flexlm
FLUENT License manager
MpCCI Flexlm

□ LMTOOLS ignores license file path environment variables

- Select in the "Service/License File" tab section the option "Configuration using Services".
- Click the "Config Services" tab section.



Configure Service

Service Name: MpCCI Flexlm

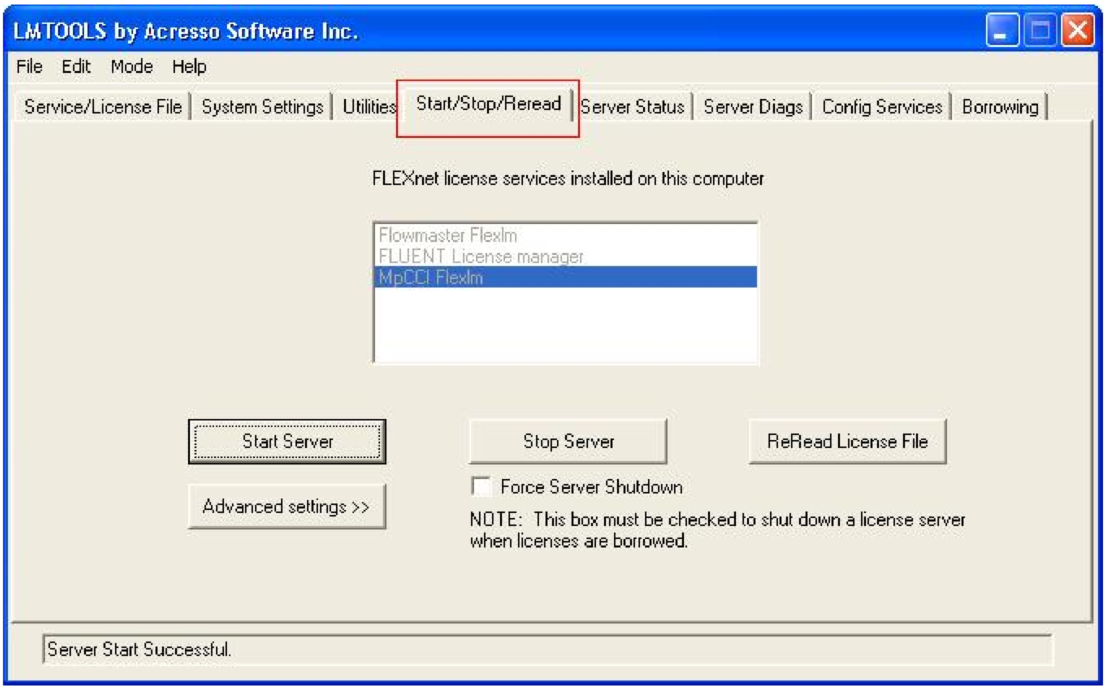
Path to the lmgrd.exe file: rkspace\mpcci\license\winxp_em64t\lmgrd.exe

Path to the license file: C:\workspace\mpcci\license\mpcci.lic

Path to the debug log file: C:\workspace\mpcci\license\mpcci.lic.log

☑ Start Server at Power Up ☑ Use Services

- Enter a service name e.g. MpCCI license manager or MpCCI FlexLM.
- Select the path of the program *lmgrd.exe* with the “Browse” button:
  < LICENSE_TOOL_INSTALLATION > /bin/lmgrd.exe
- Select the license file *mpcci.lic* with the “Browse” button:
  < LICENSE_TOOL_INSTALLATION > /license/mpcci.lic
- Activate the “Start Server at Power Up” option.
- Activate the ”Use Services“ option.
- You can optionally add a log file by providing a file name for the ”Path to the debug log file” option.
- Click on the “Save Service” button.



FLEXnet license services installed on this computer

Flowmaster Flexlm
FLUENT License manager
MpCCI Flexlm

□ Force Server Shutdown

NOTE: This box must be checked to shut down a license server when licenses are borrowed.

Server Start Successful.

- Select in the “Start/Stop/Reread” tab section the license service.
- Click on the “Start Server” button.
- The license server is now running and configured to start at power up.

3 Using the MpCCI Mapper

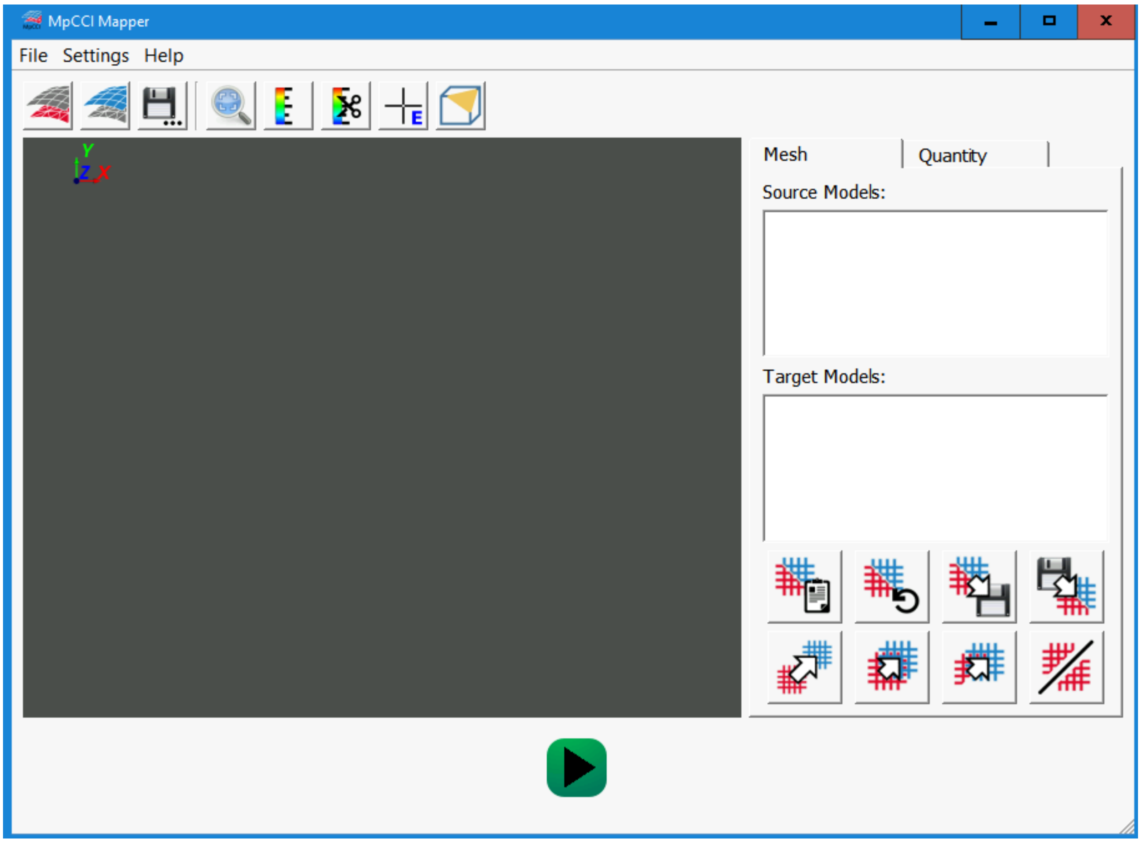


Figure 3.1: Graphical user interface of MpCCI Mapper

The graphical user interface of the MpCCI Mapper is split up in three major areas. Located on top of the application, a list of action buttons form the so called 'Toolbar' 3.2 where basic operations of the MpCCI Mapper can be accessed. Right below the toolbar, the main viewport 3.3 handles the display of model geometry and quantities in a OpenGL rendering area. Alongside of the viewport, the data panel 3.4 is devided in a mesh and a quantity related subpanel to setup a mapping step. Geometry related information, e.g. models, are shown in the 'Mesh' panel 3.4.1 where opened source and target meshes with their parts are listed. Quantity related information, e.g. the definition of mapping quantities or the validation of a quantity, can be done in the 'Quantity' panel 3.4.2. At the bottom of the window a single button is present which is responsible for starting mapping or validation.

3.1 Menus

3.1.1 File Menu

The file menu located at the top left corner of the graphical user interface 3.1. On left click on it the so called 'file menu' opens as displayed in figure 3.2 offering several options

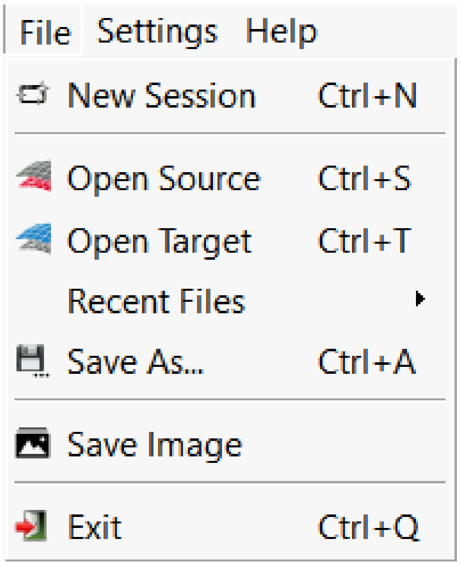


Figure 3.2: File Menu Content

for interaction with local file system. A detailed description on each option is given in table 3.1.

Entry	Shortcut	Description
New Session	Crtl+N	Delete all currently loaded models and reset program
Open Source	Crtl+S	Open a source model (often the metal forming results)
Open Target	Crtl+T	Open a target model to map to (often the crash model)
Recent Files		Last five opened source and target models can be accessed here
Save As	Crtl+A	Save a target result file. Some file formats support patching which means that the original file is read in, the new mapped values are added and a merged file is written.
Save Image	Alt+F, I	A picture from the visualisation window is captured and stored in a file. The resolution of the image is quite higher than that of the screen. Take care if you save bitmaps (size > 30Mbyte)
Exit	Crtl+Q	Closes the application

Table 3.1: Options of File Menu

3.1.2 Settings Menu

The settings menu is located at the top menu bar of the graphical user interface next to the file menu 4.2. It is designed as a drop down menu containing certain alignment, mapping algorithm and file writer configuration options.

Alignment Settings

The alignment direction of the MpCCI Mapper can be inversed in ”Settings/Alignment”. If target model is larger than the mapping source model, automatic alignment 3.4.1 tools might produce poor alignment quality. Using the inverse alignment direction, the alignment quality can be significantly improved.

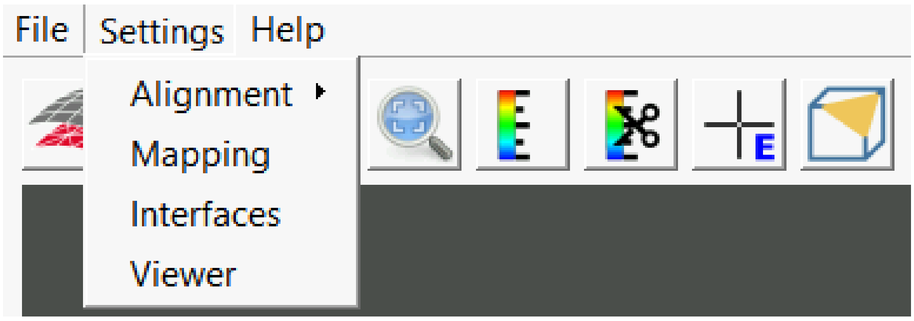
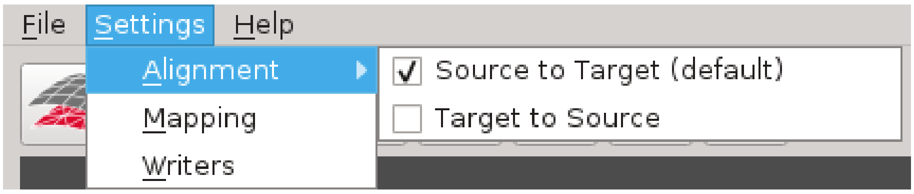


Figure 3.3: Settings Menu



☑ Source to Target (default)
□ Target to Source

Figure 3.4: Inverse alignment direction from default source on target to target on source.

Mapping Settings

MpCCI Mapper uses a k-nearest-node mapping algorithm (default k = 4) that combines robustness with data smooting. At times were averaging of certain entities is not wanted or when data smoothing shall be increased MpCCI Mapper allows to adjust the number of data points used for interpolation. This can be done in ”Settings/Mapping” as shown in figure 3.1.2 where parameter k (valid range between k=1 and k=8) can be set for explicitly quantity STRESS (and HISTORY variables for LSDyna), Material ORIENTATIONS or DEFAULT for all quantities.

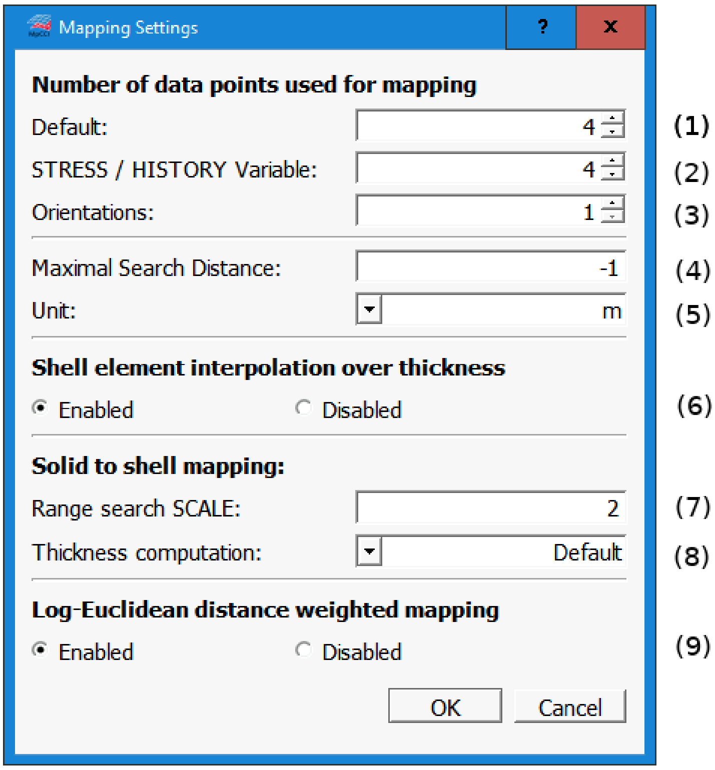
Having source and target models with different number of integration points over shell element thickness, an interpolation between the thickness integration os done. If a user does not want interpolation over element thickness, it can be disabled so the interpolated value is determined by the nearest integration type.

If the source model is a subset the target model, and data shall only be mapped within source model range, a maximal search distance can be defined.

To perform a solid to shell mapping MpCCI Mapper uses a surface to surface projection approach to determine THICKNESS information from solid geometry. As an inital step the upper and lower surfaces of the shell model get translated by half of initial shell thickness in element normal direction. In some cases (shell surface inside solid) it might be necessary to increase the shell surface translation by an addition factor. The factor “SCALE” (defaut = 1) can be defined in the correcponding line edit entry as shown in figure 3.1.2.

Interface Settings

Using the “Settings/Interface” menu entry 3.6 it is possible to configure the file readers and writers. For format LSDyna the writer can be configured to export single or multiple files. In addition, an exchange of the position and a data scaling of history variables can be defined during export. Section 4.8.6 gives an overview about how to define the ASCII-based configuration file.



Number of data points used for mapping

Default: 4

STRESS / HISTORY Variable: 4

Orientations: 1

Maximal Search Distance: -1

Unit: m

Shell element interpolation over thickness

● Enabled ○ Disabled

Solid to shell mapping:

Range search SCALE: 2

Thickness computation: Default

Log-Euclidean distance weighted mapping

● Enabled ○ Disabled

Figure 3.5: Mapping settings dialog

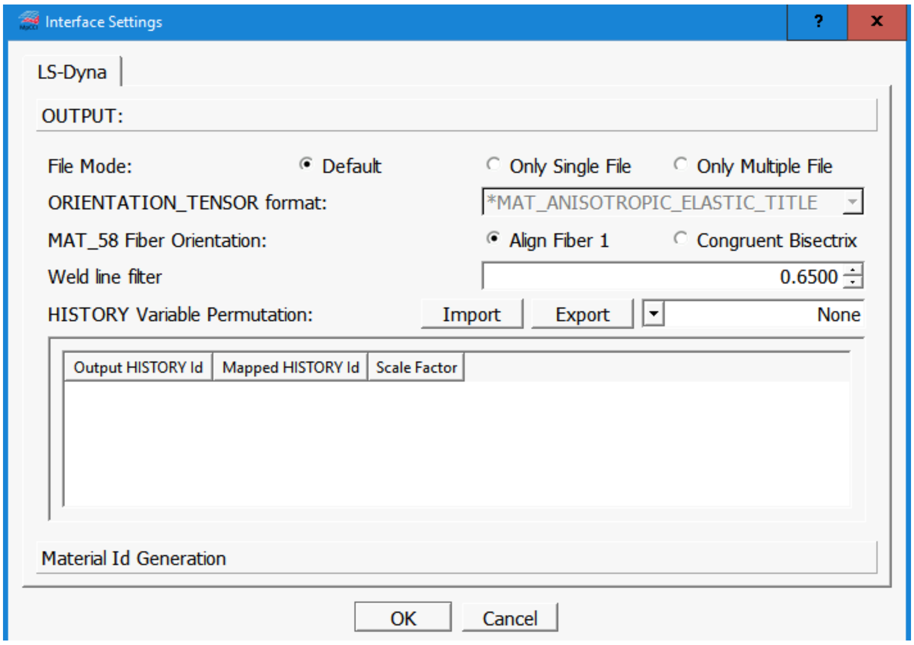
(1) Default k-nearest-node mapping algorithm parameter k
(2) Parameter k for quantity STRESS or LS-Dyna HISTORY variables
(3) Parameter k for orientations (material direction, fiber direction, ORIENTATION_TENSOR
(4) Maximal search distance for nearest-node search, negative value means unlimited
(5) Maximal search distance unit
(6) Enable or disable through thickness direction interpolation for shell elements
(7) Solid to shell mapping, translation factor of solid outer surface (default value = 2.)
(8) Solid to shell mapping, thickness computation method (Default: solid outer surface mode, Experimental: top and bottom surface mapped from target)
(9) Enable or disable distance weighting for Log-Euclidean metric mapping algorithm (ORIENTATION_TENSOR only)

Viewer Settings

Using the "Settings/Viewer" menu entry 3.7 it is possible to configure the appearance of the 3D viewport. Here, the viewport background color as well as the scalar bar color map can be configured.

3.1.3 Help Menu

The help menu gives information about MpCCI Mapper version and has an easy access to this documentation by pressing F1.



File Mode: ● Default ○ Only Single File ○ Only Multiple File

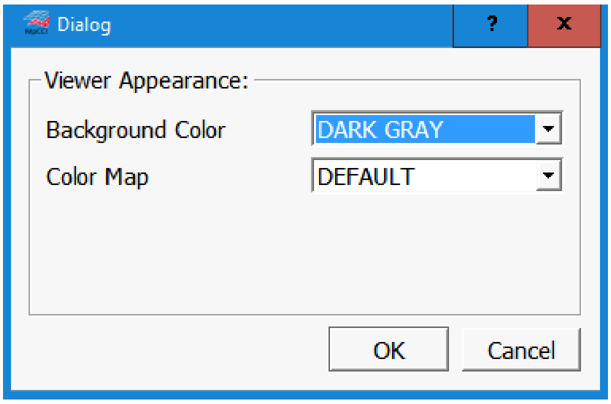
ORIENTATION_TENSOR format: *MAT_ANISOTROPIC_ELASTIC_TITLE

MAT_58 Fiber Orientation: ● Align Fiber 1 ○ Congruent Bisectrix

Weld line filter 0.6500

HISTORY Variable Permutation: None

Figure 3.6: Interface settings dialog



Background Color DARK GRAY

Color Map DEFAULT

Figure 3.7: Viewer settings dialog

3.2 Toolbar

The MpCCI Mapper toolbar is located at the top of the graphical user interface. It offers functionality for view navigation as well as shortcut buttons to file import and export of the file dialog. Figure 3.8 gives an overview on possible actions.

	Load a new source model from file
	Load a new target model from file
	Save active (visible) target models
	Reset Camera to show all active models
	Adjust the quantity color range by the user or reset to automatic range
	Clip a scalar value range Values above and below new limits are set to lower resp. upper limit
	Show info about selected element 3.5.1
	Slice through model and plot scalars along traverse 3.5.2

Figure 3.8: Toolbar button description

3.3 Viewport

The viewport of the graphical user interface is the area where loaded models are displayed. Herein the user can interactively inspect an arbitary number of models at the same time. Descriptions for use of the viewport by mouse and hotkeys are given in sections 3.3.1 and 3.3.2;

3.3.1 Basic View Interaction

Spinning of models	Press and hold left mouse button on a model, then move mouse to rotate
Translating models	Press and hold middle mouse button on a model, then move mouse to desired location
Zooming models	Press and hold right mouse button on a model, then move mouse upwarts to zoom in or downwards to zoom out
Rotate view	Press and hold left mouse button on background or hold ’Shift’ button
Translating view	Press and hold middle mouse button on background or hold ’Shift’ button
Zooming view	Press and hold right mouse button on background or hold ’Shift’ button

Table 3.2: View and model interaction

3.3.2 Viewport Hotkeys

Remark: To change the style of only some parts deactivate the others, switch the style and activate the others again.

’w’	Wireframe mode: active parts are presented as wireframes
’s’	Surface mode: active parts are presented as surfaces
’b’	Surface with mesh wireframe (’w’+’s’)
’p’	Points mode: active parts are presented as point clouds
’o’	Opaque mode: toggle opacity/transparency of models
’Shift’	Disable model interaction on rotation, translation and zoom

Table 3.3: List of hotkeys for viewport interaction

3.4 Data Panel

The data panel organizes geometry and quantity releated information. It is devided in a ’Mesh’ section 3.4.1 where opened source and target meshes with their parts are listed and a ’Quantity’ section 3.4.2 where quantities for mapping and validation are managed.

3.4.1 Mesh Panel

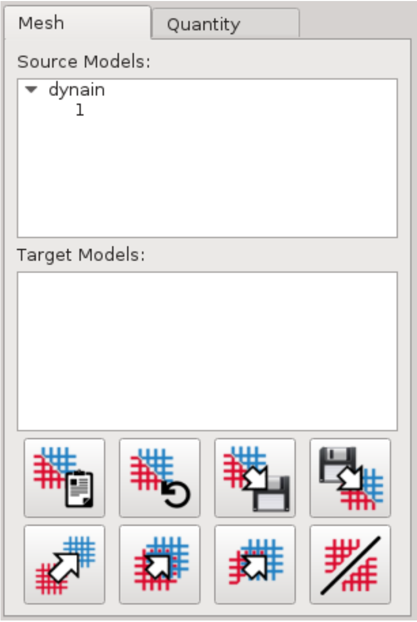


Figure 3.9: Mesh Panel

The ’Mesh’ panel can be used to manage models as well as their relative postition to each other. Here, each loaded model is assigned to the list of source or list of target models. For each model, a so call model tree gives an overview about the model and its subset partitions. Single parts or models can be activated or deactivated for visualization and an unit system can be assigned for each model separately.
At the bottom of the ’Mesh’ panel, a form based layout offers different automatic model alignment options 3.4.1 which can be used to place models before data exchange.

Model and Part Selection

Double clicking on model name
Models are activated or deactivated (for mapping, aligning and visualization)

Double clicking on part name
Parts are activated or deactivated (for mapping, aligning and visualization)

Click on triangle
Collapse or expand the parts tree

Right click on model name
Pop up of Models Context Menu

Table 3.4: Part selection options

Models Context Menu

On right click on model tree the 'models context menu' opens.

Automatic Positioning

A common problem in data mapping between different simulation models is the use of varying local coordinate systems. Here, the models might not have the same spatical location in a common coordinate system. Hence, for a position-dependent neighborhood computation, both models need to be aligned in a preprocessing step. This alignment step can be done using the automatic positioning features in MpCCI Mapper (c.f. table 3.12).

If the transformation of both coordinate systems are know, a manual setup of the MpCCI Mapper transformation file can be done in he following way:

Example Transformation

If $A = (a_{ij})_{i,j=1,..,3}$ is the rotation matrix of a model in the 3-dimensional space and $T = (t_i)_{i=1,..,3}$ the translation vector we have the transformation Q in homogenous coodinates

$$Q = \begin{pmatrix} a_{11} & a_{12} & a_{13} & t_1 \\ a_{21} & a_{22} & a_{23} & t_2 \\ a_{31} & a_{32} & a_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
*transformation =  -3.289344e-05  -1.000000e-00   1.079831e-06   2.674954e-04
                    5.307190e-04   1.062373e-06   9.999999e-01  -3.006939e-01
                   -9.999999e-01   3.289400e-05   5.307189e-04   1.028435e+00
                    0.000000e+00   0.000000e+00   0.000000e+00   1.000000e+00
```

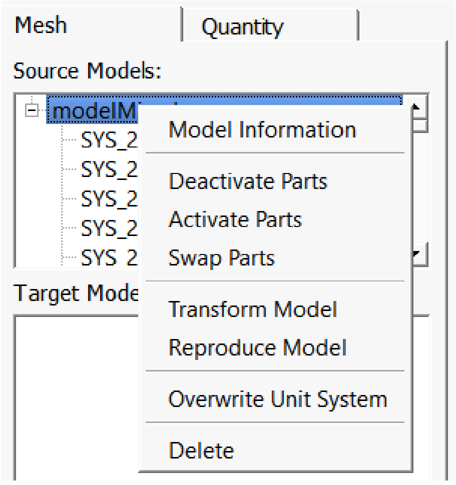


Figure 3.10: Models Menu

Model Information
Shows Information about the mesh topology and quantities

Deactivate Parts
Deactivates all parts of model (for mapping, aligning and visualization)

Activate Parts
Activate all parts of model (for mapping, aligning and visualization)

Swap Parts
Deactivates all activated and activates all deactivated parts

Transform Model
Opens transform model dialog to translate or scale mesh individually

Reproduce Model
Opens replicate model dialog to transform symmetric meshes to full meshes or duplicate meshes by fixed translation

Project onto Source
Available on target model only. If source and target model consist of shell elements a geometry projection of target geometry to source shape can be performed. REMARK: Recommended only for small displacements.

Overwrite Integration Rule
Available on LSDyna target models only. Change element formulation and number of integration points though thickness before mapping.

Overwrite Unit System
During reading all data (point coordinates, thickness, stress..) is transformed to SI-units. During writing all data is transformed back to the original unit system. Use this option if the first guess of the unit system was wrong.

Delete
Removes mesh from the GUI and database

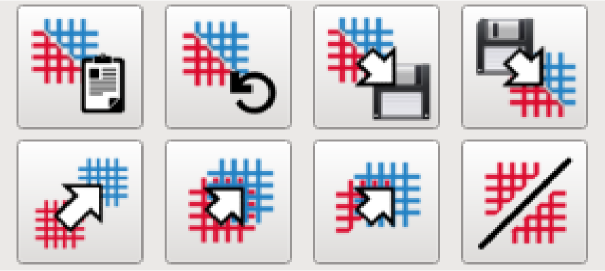


Figure 3.11: Automatic model alignment tools

	Remember current position for mapping and validation
	Set model positions to last remembered position.
	Rough Align: Used to bring equal meshes close to each other.
	Fine Align: Align meshes for mapping. Can be used several times to improve the result.
	Symmetric Align: Used if one of the models uses axes symmetry
	Mirror Geometry: Mirrors all active models Warning: Mirroring is applied to the geometry. When saving the mapping results of a mirrored target, the mirrored geometry is written to the output file.
	Save current positions to a file: 1. It is possible to store the transformations in Mapper format (*.trf) 2. Export target mesh nodes in current postition in LSDyna keyword format 3. Export source mesh nodes in current postition in LSDyna keyword format 4. Export target mesh position using INCLUDE_TRANSFORM LSDyna card 5. Export source mesh position using INCLUDE_TRANSFORM LSDyna card
	Load transformation from a file: It is possible to read the transformation of a target to a source model in MpCCI Mapper format. Transformation needs to be initialized with keyword "*transformation". Declation of the transformation begins with a new line and is read linewise; entries need to be separated by blanks. Note: All active targets will be transformed.

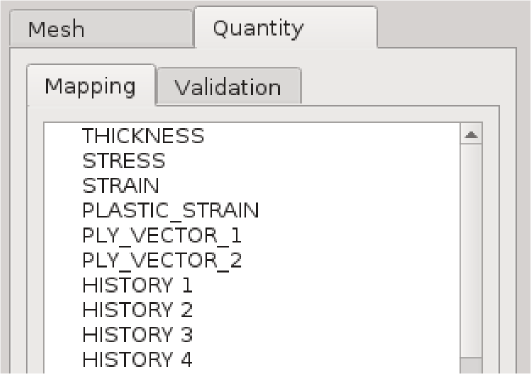
Figure 3.12: List of available automatic positioning methods.

3.4.2 Quantity Panel

The 'Quantity' panel, as second entry of the data panel, lists up quantities which are currently loaded inside MpCCI Mapper and are available for mapping to other models or for validation purpose.

Mapping Quantity Selection

In the 'Mapping' section (c.f. figure 3.13) of the quantity panel a list of all available quantities is shown. By default, all present quantities are set active for mapping. If only a subset should be used, unwanted transfer quantities can be deactivated in the list overview by direct user interaction 3.4.2 or semi-automatic using the quantity context menu 3.4.2.



THICKNESS
STRESS
STRAIN
PLASTIC_STRAIN
PLY_VECTOR_1
PLY_VECTOR_2
HISTORY 1
HISTORY 2
HISTORY 3
HISTORY 4

Figure 3.13: Selection of quantites for mapping

Left Click
Select/deselect quantity for visualization of values as contour plot

Double (Left or Mid) Click
Activate/deactivate quantity for mapping (black/grey letters)

Right Click
Open quantity dialog box 3.4.2

Deselect
Show parts

Table 3.5: Mapping quantity interaction

Remark:

Averaged values of all element integration points are used for element based data visualisation. This may lead to varying color display if source and target have different number of integration points.

Quantity Context Menu

By right clicking on a quantity name the quantity context menu opens. Here, automatic selection and deselection operations on quantities can be done or additional information on especially tensor data can be requested for display (c.f. table 3.4.2).

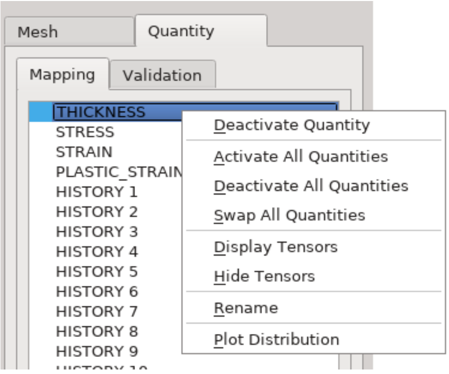


Figure 3.14: Overview of the Quantity Context Menu

Activate
Activate quantity for mapping

Deactivate
Deactivate quantity for mapping

Activate All Quantities
Activates all defined quantities for mapping

Deactivate All Quantities
Deactivates all quantities

Swap All Quantities
Activates currently deactivated quantities, deactivates currently activated quantities

Display Tensors
Show principal axes of a tensor quantity

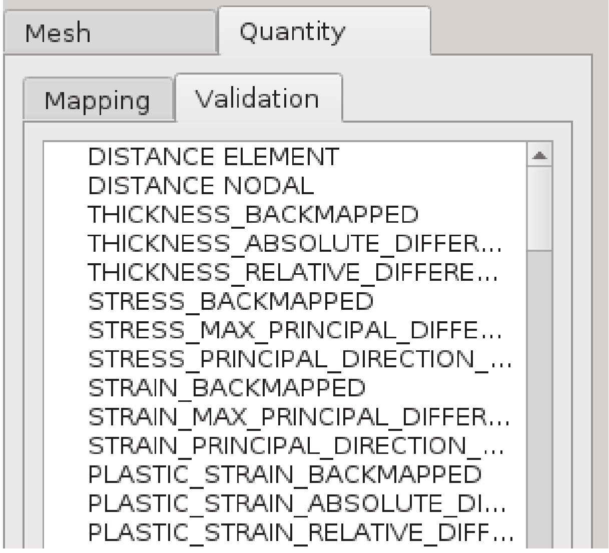
Hide Tensors
Hides principal axes of a tensor quantity

Rename
Assign a new name for selected quantity

Plot Distribution
Shows a percentual distribution of selected quantity per mesh in a 2D plot 3.5.3

Table 3.6: Quantity Dialog Options

3.4.3 Quantity Validation



DISTANCE ELEMENT
DISTANCE NODAL
THICKNESS_BACKMAPPED
THICKNESS_ABSOLUTE_DIFFER...
THICKNESS_RELATIVE_DIFFERE...
STRESS_BACKMAPPED
STRESS_MAX_PRINCIPAL_DIFFE...
STRESS_PRINCIPAL_DIRECTION_...
STRAIN_BACKMAPPED
STRAIN_MAX_PRINCIPAL_DIFFER...
STRAIN_PRINCIPAL_DIRECTION_...
PLASTIC_STRAIN_BACKMAPPED
PLASTIC_STRAIN_ABSOLUTE_DI...
PLASTIC_STRAIN_RELATIVE_DIFF...

Figure 3.15: Quantity Validation Panel

The Validation panel is designed for validating mapped and non-mapped quantity data on given source and target meshes. By default all quantities defined on source and target model will be used during validation process. For individual configuration of quantities use the right click context menu 3.16.
Note: Validation functionality can also be used to compare nonmapped quantities.

DISTANCE ELEMENT
Shows the association distance of the current mapping position. The distance of the visible target parts to the visible source parts is calculated.

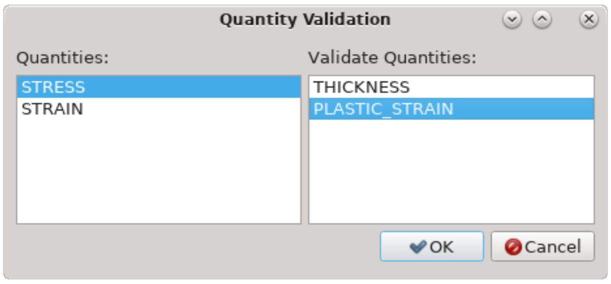
DISTANCE NODAL
Shows the distance to the next node of the current mapping position. The distance of the visible target parts to the visible source parts is calculated.

Scalar Quantities
During validation of a scalar quantity three validation quantities are created. First, a quantity containing the result of mapping information back from target to source mesh is generated having extension "_BACKMAPPED". Second, using base and "_BACKMAPPED" quantity, absolute difference values are computed and stored having "_ABSOLUTE_DIFFERENCE" extension. Third, using base and "_ABSOLUTE_DIFFERENCE" quantity absolute relative difference values are computed and stored having "_RELATIVE_DIFFERENCE" extension.

Vector Quantities
During validation of a scalar quantity three validation quantities are created. First, a quantity containing the result of mapping information back from target to source mesh is generated having extension "_BACKMAPPED". Second, using base and "_BACKMAPPED" quantity, absolute difference values are computed per vector component and stored having "_ABSOLUTE_DIFFERENCE" extension. Third, using base and "_BACKMAPPED" quantity, the angular deviation in degree is computed and stored having "_DIRECTION_DIFFERENCE" extension.

Tensor Quantities
During validation of a scalar quantity three validation quantities are created. First,

a quantity containing the result of mapping information back from target to source mesh is generated having extension “_BACKMAPPED”. Second, using base and “_BACKMAPPED” quantity, absolute difference of tensor principal values are computed and stored having “_MAX_PRINCIPAL_DIFFERENCE” extension. Third, using base and “_BACKMAPPED” quantity, the angular deviation in degree is computed for all principal directions and stored having the extension “_PRINCIPAL_DIRECTION_DIFFERENCE”.



Quantities:

- STRESS
- STRAIN

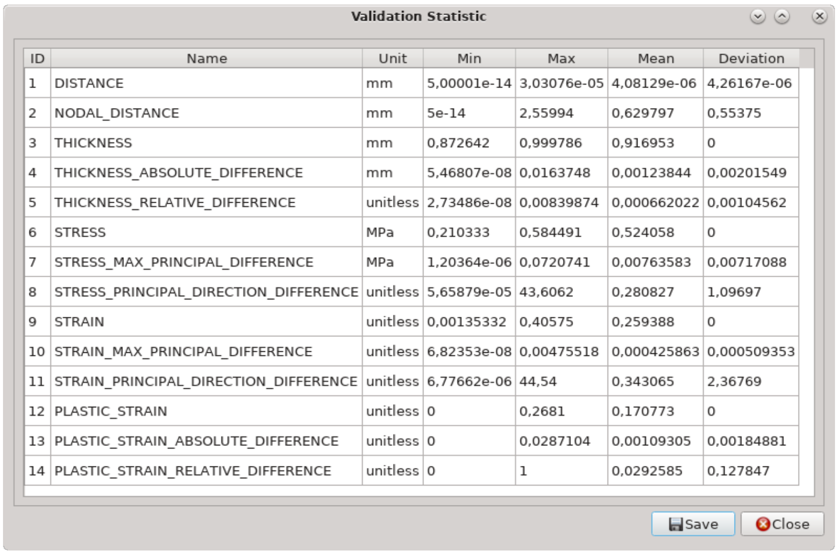
Validate Quantities:

- THICKNESS
- PLASTIC_STRAIN

Figure 3.16: Quantity Validation Dialog

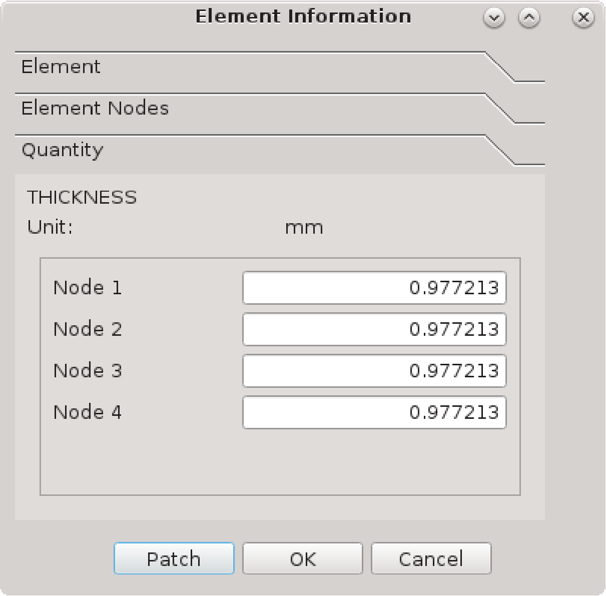
Supplementary to the automatic mode, individual quantity sets can be defined using right click optin on validation list and selecting “Validate Quantity” from context menu. In the following “Quantity Validation Dialog” 3.16 available quantities can be assigned to validation list.

After the (mapping) validation has been performed, a statistical analysis of the obtained validation quantities can be displayed. Here, for each validation quantity the absolute minimum and maximum as well as the arithmetic mean together with the standard deviation is computed. At the end, those data is visualized in a tabule report dialog 3.17. Optionally the table result can be exported as comma separated value (*.csv) file.



ID	Name	Unit	Min	Max	Mean	Deviation
1	DISTANCE	mm	5,00001e-14	3,03076e-05	4,08129e-06	4,26167e-06
2	NODAL_DISTANCE	mm	5e-14	2,55994	0,629797	0,55375
3	THICKNESS	mm	0,872642	0,999786	0,916953	0
4	THICKNESS_ABSOLUTE_DIFFERENCE	mm	5,46807e-08	0,0163748	0,00123844	0,00201549
5	THICKNESS_RELATIVE_DIFFERENCE	unitless	2,73486e-08	0,00839874	0,000662022	0,00104562
6	STRESS	MPa	0,210333	0,584491	0,524058	0
7	STRESS_MAX_PRINCIPAL_DIFFERENCE	MPa	1,20364e-06	0,0720741	0,00763583	0,00717088
8	STRESS_PRINCIPAL_DIRECTION_DIFFERENCE	unitless	5,65879e-05	43,6062	0,280827	1,09697
9	STRAIN	unitless	0,00135332	0,40575	0,259388	0
10	STRAIN_MAX_PRINCIPAL_DIFFERENCE	unitless	6,82353e-08	0,00475518	0,000425863	0,000509353
11	STRAIN_PRINCIPAL_DIRECTION_DIFFERENCE	unitless	6,77662e-06	44,54	0,343065	2,36769
12	PLASTIC_STRAIN	unitless	0	0,2681	0,170773	0
13	PLASTIC_STRAIN_ABSOLUTE_DIFFERENCE	unitless	0	0,0287104	0,00109305	0,00184881
14	PLASTIC_STRAIN_RELATIVE_DIFFERENCE	unitless	0	1	0,0292585	0,127847

Figure 3.17: Quantity Validation Report



THICKNESS
Unit: mm

Node 1	0.977213
Node 2	0.977213
Node 3	0.977213
Node 4	0.977213

3.5 Postprocessing

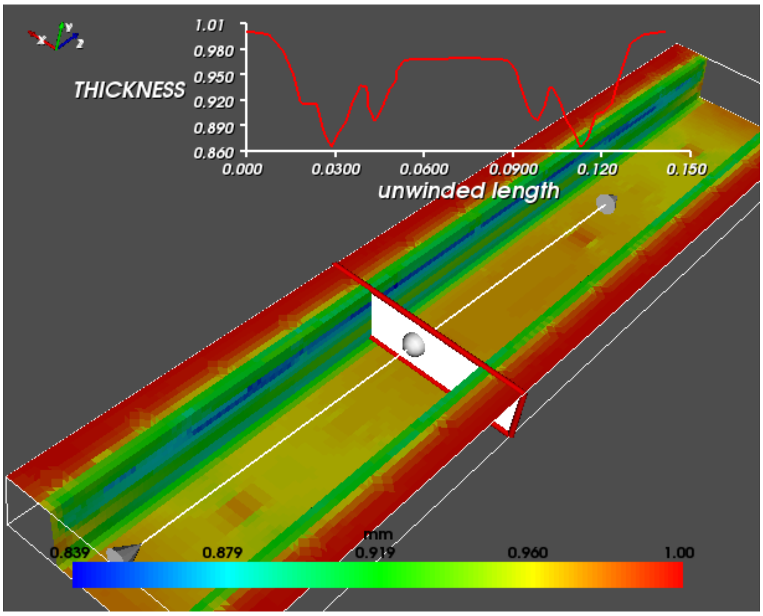
3.5.1 Element Information

By pressing the ”element information button” located at the toolbar and picking an element of a mesh in the viewport it is possible to access geometric and (scalar) quantity information of this element. As graphical assistance the selected element in bordered in the visualisation window. The new information window lists:

- Id of selected element
- Type of selected element
- Ids of nodes forming the element
- Scalar value on element / each node are show and can be edited
- Current unit of active quantity

3.5.2 Slice Model and Plot

Pressing the "slice model plot“ of the toolbar an interactive plane widget will be shown in the viewport. By manual interaction with the plane normal arrow, the position and direction of a slice through the model can be defined.



x-axis: unwinded length of the traverse
y-axis: interpolated quantity value

Basic interactions on plane widget

- Recompute values by click on plane surface
- Rotate plane normal by left click on arrow and mouse movement
- Translate normal rotation point by middle click on ball and manual positioning
- Translate plane by either left or middle click on plane
- Shrink or extend bounding box by right click and mouse movement

For every position update of the plane surface the MpCCI Mapper tries to compute a so called length plot corresponding to the quantity distribution. Thereby element edges are intersected with the plane representation and nodal or elemental scalar data is interpolated on the intersection points. Then a traverse is computed that represents the geometry alongthe defined plane.

Notes and restriction

Multiple models can simultaniously be cut using the slice functionality; multiply plots get an x-axis offset corresponding to the geometric distance of the traverse starting point. Plot axis can get manually scaled at the end of the axis (cursor changes).

3.5.3 Quantity Plot

To study the relative distribution of a quantity, it's distribution on a percentage basis can be visualized as 2D-plot. First, select desired quantity in "Mapping“ or "Validation“ tab of the quantity panel. On right-click the quantity name the context menu opens 3.4.2, then select ”Plot Quantity“ option in list.

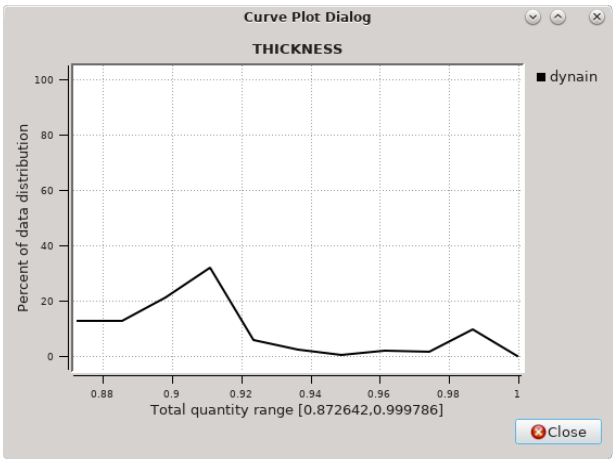


Figure 3.18: Distribution of a certain quantity

The value range interval will be devided in ten equally spaced sub-intervals and the relative frequency of data points in the sub-interval range is plotted for each model separately.

3.6 Mapping - Theory of Algorithms in MpCCI Mapper

The discrete mapping problem for a set of N known data points, described as a list of tuples $[(x_1, u_1), (x_2, u_2), ..., (x_N, u_N)]$, where x_i denote points in the 3-dimensional space and u_i the corresponding values of an unknow function $u(x) : x \to \mathbb{R}, \quad x \in \mathbf{D} \subset \mathbb{R}^3$ Goal of the interpolation problem is to find a function $\tilde{u}$ to be ”smooth“ and to be exact $\tilde{u}(x_i) = u_i$.

3.6.1 Euclidean inverse distance weighted k-nearest

A general form of finding an interpolating function $\tilde{u}$ at a given point $\mathbf{x}$ based on samples $\tilde{u}(x_i) = u_i$ for i = 1,..., N is the Euclidean Inverse Distance Weighting method

$$\tilde{u}(\mathbf{x}) = \begin{cases} \frac{\sum_{i=1}^{N} w_i(\mathbf{x}) u_i}{\sum_{i=1}^{N} w_i(\mathbf{x})}, & \text{if} \quad ||x - x_i|| > 0 \quad \text{for all i,} \\ u_i, & \text{if} \quad ||x - x_i|| = 0 \quad \text{for some i} \end{cases}$$

where

$$w_i(\mathbf{x}) = \frac{1}{||x - x_i||^p}$$

and p is a positive real number, called the power parameter. As the interpolation function depends $\tilde{u}$ on all data tuples, its evaluation gets more and more time consuming with increasing number of N. By construction of Inverse Distance Weighting, the influence - meaning the weight - of one data pair (x_i, u_i) decreases exponentially with the Euclidean distance to x_i.

Utilizing the exponential decrease of the weighting function, the above interpolation scheme can be modified by the k-nearest approach

$$w_i(\mathbf{x}) = \begin{cases} \frac{1}{||x - x_i||^p}, & \text{for those indexes i having the k-th minimal distance to } x \\ 0, & \text{else} \end{cases}$$

3.6.2 Log-Euclidean weighted k-nearest

The **exponential** of a matrix A is defined by

$$exp(A) = \sum_{i=1}^{\infty} \frac{A^n}{n!}.$$

To a given matrix A, another matrix B is said to be the **matrix logarithm** of A if $exp(B) = A$ is satisfied.
Log-Euclidean metric for symmetric positive definite (spd) matrices

$$dist(S_1, S_2) = ||log(S_1) - log(S_2)||$$

builds an isomorphism (the algebraic structure of a space is conserved) and is an isometry (distances are conserved) between the space of spd matrices and the Euclidean space. Having a set of N tensors $S_1, ..., S_N$ with arbitary positive weights $w_1, ..., w_N$ we can fomulate the Log-Euclidean Fréchet mean mean of N tensors by

$$\mathbb{E}(S_1, ..., S_N) = exp(\sum_{i=1}^{N} w_i log(S_i))$$

A comparison of the Euclidean and the Log-Euclidean interpolation result of tensors is illustrated in figure 3.19 [1].

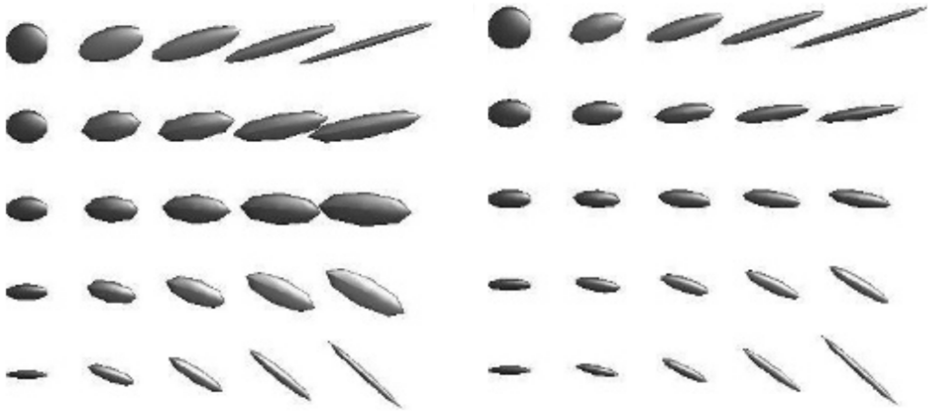


Figure 3.19: Bilinear interpolation of 4 tensors at the cornersof a grid. Left figure: Euclidean interpolation Right figure: Log-Euclidean interpolation.

[1] Images from Fast and Simple Calculus on Tensors in the Log-Euclidean Framework, Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache

3.7 Mapping with Mapper

A complete mapping cycle between one source and one target model can be done with just a few clicks. The following complete mapping and validation workflow illustrates the typical use of MpCCI Mapper tool. Most of the described steps are optional and may depend on the models and detail of mapping analysis.

Complete Mapping and Validation Workflow

1. Load a source mesh
2. Load a target mesh
3. Switch to mesh panel (optional)
4. Select source and target mesh parts that shall be used for mapping (optional)
5. Align models (optional)
6. Switch to quantity mapping panel (optional)
7. Select quantities which shall be mapped on target (optional)
8. Press mapping button
9. Switch to quantity validation panel (optional)
10. Press validation button for quantity validation (optional)
11. Save target to file

Restrictions:

1. Only quantities are mapped that are available on the source and can be created (if not present) on target.
2. No iterative mapping is supported. If you want to map several sources on one target you need to map all at the same time.
3. Mapping solid to shell models requires only one source and one target mesh.
4. Integral values cannot be conserved.
5. Out of plane integration rules are defined for Gauss up to 16, Lobatto up to 16, Trapezodial up to 15, Simpson up to 9 integration points.
6. For stresses only nearest data point should be used (c.f. Mapping Parameter Section 3.1.2).

3.8 Batch Mode

MpCCI Mapper allows direct loading, mapping of models and result export via program argument list. The required file readers are automatically detected by file extension.
By default the mapping in batch mode does require same local coordinate system for source and target model. In cases where the automatic alignment tools of MpCCI Mapper (c.f. 3.4.1) produces an acceptable alignment, these automatic positioning can be used in batch as well. If automatic alignment does not work, it is possible to use exported model transformation from manual alignment.
List of MpCCI Mapper argument list options:

-nogui
Enable batch mode of MpCCI Mapper

-csv2dyna
Use Autoform CSV output to LSDyna keyword conversion tool

-csv2vtk
Use CSV to VTK legacy conversion tool

-disablestress
Disable mapping of quantity STRESS

-maximalsearchdistance
Define the maximal search distance parameter (in model length unit) (c.f. 3.1.2)

-configfile
Option to define program settings based on configuration file (c.f. 3.8.5)

-help
This screen

List of MpCCI Mapper arguments for automatic alignment:

-align
Enable automatic (coarse + fine) model alignment

-coarsealign
Enable coarse model alignment

-finealign
Enable fine model alignment

-inversealign
Enable automatic (coarse + fine) model alignment interchanging source and target model

-inversecoarsealign
Enable coarse model alignment interchanging source and target model

-inversefinealign
Enable fine model alignment interchanging source and target model

-applytransform
Enable model alignment using predefined transformation matrix (c.f. 3.4.1)

3.8.1 Batch on Linux Cluster

The graphical user interface of MpCCI Mapper is based on the QT GUI-Toolkit. Its implementation on Linux does require an active X Window System framework even if the program is running in batch. Normally, Linux cluster compute nodes do not allow the execution of graphical applications using X Window System. Therefore exist virtual display server, e.g. Xvfb (X virtual framebuffer), implementing the X11 display server protocol. Xvfb performs all graphical operations in virtual memory without showing any screen output.
Before execution of MpCCI Mapper in a shell script on a cluster compute node, a virtual display server can be started by adding the lines

```
Xvfb :1 &
export DISPLAY=:1
```

previous to the `mapper.exe` program call.

3.8.2 Examples

MpCCI Mapper examples using automatic model loading, mapping and result export:

1. `mapper.exe source_file`
Starts GUI and loads `source_file` as source model into MpCCI Mapper .

2. `mapper.exe source_file target_file`
Starts GUI and loads `source_file` as source model and `target_file` as target model into MpCCI Mapper and maps all defined quantities from source to target.

3. `mapper.exe source_file target_file result_file`
Same as 2); the mapped results are then exported into `result_file` in the default export format of `target_file`.

3.8.3 Batch Mode Example

To perform an automatic model import and mapping in batch MpCCI Mapper add the `-nogui` option as first argument:

4. `mapper.exe -nogui source_file target_file result_file`
Same as 3) but the MpCCI Mapper GUI is disabled during the whole process.

5. `mapper.exe -nogui -config config_file source_file target_file result_file`
Same as 4) but the MpCCI Mapper GUI parameter are set using configuration file.

3.8.4 Alignment Examples

To make use of automatic positioning add one of the options `-align`, `-coarsealign`, `-finealign` or their inverse to argument list, e.g

2b. `mapper.exe -align source_file target_file`

3b. `mapper.exe -finealign source_file target_file result_file`

4b. `mapper.exe -nogui -align source_file target_file result_file`

3.8.5 Batch Configuration File

Beside starting the MpCCI Mapper in batch with additional process control commands, the complete mapping setup can be specified using a text based configuration file.

***optionAlign**

Define automatic model alignment method (same as the options in 3.8)

Value	Description
none	Models are aligned (Default)
-align	Enable automatic (coarse + fine) model alignment
-coarsealign	Enable coarse model alignment
-finealign	Enable fine model alignment
-inversealign	Enable automatic (coarse + fine) model alignment interchanging source and target model
-inversecoarsealign	Enable coarse model alignment interchanging source and target model
-inversefinealign	Enable fine model alignment interchanging source and target model

***optionTransformation**

Define transformation file for model alignment (same as "-applytransform" in 3.8)

Value	Description
none	default
Global_Path_To_File/transform.trf	

***optionMappingDefault**

Define parameter "Default" in Mapping Settings 3.1.2

Value	Description
4	Default
1-8	Specify number of data points to use for interpolation

***optionMappingStressOrHistory**

Define parameter "STRESS/HISTORY" in Mapping Settings 3.1.2

Value	Description
4	Default
1-8	Specify number of data points to use for interpolation

***optionMappingOrientations**

Define parameter "Orientations" in Mapping Settings 3.1.2

Value	Description
1	Default
1-8	Specify number of data points to use for interpolation

***optionMappingMaximalSearchDistance**

Define parameter "Maximal Search Distance" in Mapping Settings 3.1.2

Value	Description
-1	No maximal search distance (Default)
$d > 0.$	Maximal search distance d should be used

***optionMappingMATFEMSettings**

Enable special handling of LSDyna history variables when mapping a MATFEM GenYld - CrachFEM result file

Value	Description
0	Disabled (Default)
1	Enabled

***optionReaderDynaMAT249**

Active automatic transformation of ply orientation for history variables of LSDyna MAT_249 4.8.4

Value	Description
0	Disabled (Default)
1	Enabled

***optionWriterDynaOutput**

Define the mode output files for LSDyna are written

Value	Description
0	Single and multiple file are written (Default)
1	Single file only is written
-1	Multiple files only are written

***optionWriterDynaMAT58FiberMode**

Automatic transformation of ply orientation into MAT_58 4.8.4

Value	Description
0	Align element coordinate system using fiber 1 (Default)
1	Align element coordinate system congurent bisectrix

***optionWriterDynaOrientationToHistory**

Define how variable ORIENTATION_TENSOR from molding simulation is handled on LSDyna file export

Value	Description
0	Disabled (Default), ELEMENT_SOLID_ORTHO is written from ORIENTATION_TENSOR 4.8.5
1	Enabled, generate MATFEM GenYld - CrachFEM history variables

***optionDiableStress**

Special flag to disable quantity STRESS for mapping

Value	Description
0	Map quantity STRESS (Default)
1	Disable quantity STRESS for mapping

***optionMappingQuantity**

Define a list of quantities to map from source to target

Value	Description
-1	Map all quantities (Default)
#N ”Quant_1“ ...	Map only the N quantities defined by named list

Example of a configuration file to map only ”THICKNESS“ and ”PLASTIC_STRAIN“ using automatic model alignment:

```
#comment line
*optionAlign
-align
*optionTransformation
none
*optionDiableStress
0    # 0 = off  1 = on
```

```
*optionMappingDefault
4     # 1 to 8 points
*optionMappingStressOrHistory
4     # 1 to 8 points
*optionMappingOrientations
1     # 1 to 8 points
*optionMappingMaximalSearchDistance
-1.   # Only relevant if value greater 0.
*optionMappingThicknessInterpolation
1     # 0 = off  1 = on
*optionMappingMATFEMSettings
0     # 0 = off  1 = on
*optionReaderDynaMAT249
0     # 0 = off  1 = on
*optionWriterDynaOutput
1    # [-1 / 0 / 1]  (Split File / Split and Combined Files / Combined File)
*optionWriterDynaMAT58FiberMode
1     # 0 = align X  1 = congruent
*optionWriterDynaOrientationToHistory
1     # 0 = off  1 = on
*optionMappingQuantity
2 "THICKNESS" "PLASTIC_STRAIN"
```

4 Solver Formats

4.1 Abaqus

4.1.1 File Format

Abaqus Input Deck (*.inp) is supported for reading and writing.

The model can consist of serveral *PART definitions but requires unique node id assignments. For example having two *PART definitions "A", nodes numbered from 1 to 100, and "B", nodes numbered from 1 to 80, will lead to an error due to double defined node ids.

Thus, a renumbering of node ids can be performed using Abaqus CAE.

- Import your model file in Abaqus CAE
- Create a new Job from imported model
- Right-Click on the created Job entry and select 'Write Input'

In MpCCI Mapper multiple instances of a *PART is not supported.

An element-based *SURFACE defintion is required for surface coupling.
Abaqus/CAE: Create a surface using the Surfaces tool. See also "13.7.6 Using sets and surfaces in the Assembly module" in the Abaqus/CAE User's Manual. You can also use surfaces defined in the Part module.
Input File: A surface is created with `*SURFACE, NAME=<surface name>, TYPE=ELEMENT`, see section 2.3.3 Defining element-based surfaces" of the Abaqus Analysis User's Manual.

4.1.2 Element Types

Shells

MpCCI Mapper supports reading the following shell element types: S4, S4R, S4RS, S4RS, S3, S3R, S3RS

Solids - Surface Definition

MpCCI Mapper supports reading *SURFACE definition of the following 3D solid elements: DC3D4, C3D4, DC3D5, C3D5, DC3D6, C3D6, DC3D8, C3D8, DC3D10, C3D10

4.1.3 Transferable Quantities

Quantity	Read	Write
THICKNESS	x	x
STRESS	x	x
PLASTIC_STRAIN	x	x
PRESSURE	-	x
ORIENTATION	-	x
PLY_VECTOR	-	x

Quantity	Boundary Condition Type
THICKNESS	`*NODAL THICKNESS`
	`*DISTRIBUTION,LOCATION=ELEMENT,NAME=SHELL_THICKNESS_DIST`
STRESS	`*INITIAL CONDITIONS,TYPE=STRESS`
PLASTIC_STRAIN	`*INITIAL CONDITIONS,TYPE=HARDENING`
PRESSURE	`*DLOAD`
ORIENTATION	`*ORIENTATION`
	`*DISTRIBUTION TABLE` using `COORD3D`, `*DISTRIBUTION`
PLY_VECTOR	`*SHELL SECTION, , COMPOSITE`
	`*DISTRIBUTION TABLE`, `*DISTRIBUTION` and `*ORIENTATION`

4.1.4 Output Files

To use the files in your input file include them via the ”INCLUDE” keyword. To register the nodal thickness add the ”NODAL THICKNESS” option to the relevant ”SHELL SECTION” entries
e.g.:

```
*SHELL SECTION,NODAL THICKNESS,ELSET=...,SECTION INTEGRATION=SIMPSON,MATERIAL=...
```

To register element thickness use the ”DISTRIBUTION TABLE” option, e.g.

```
*SHELL SECTION,SHELL THICKNESS=SHELL_THICKNESS_DIST,ELSET=...,SECTION INTEGRATION=...
```

***.inp**
File with model geometry information Include statements for other result files

***.strain**
Equivalent plastic strain as *INITIAL CONDITIONS,TYPE=HARDENING

***.stress**
Stress tensor as *INITIAL CONDITIONS,TYPE=STRESS

***.thick**
Shell THICKNESS as nodal thickness

***.elemthick**
Shell THICKNESS as element based thickness

***.pressure**
Element surface pressure as *DLOAD

***.orientation**
Element local material directions for anisotropic materials

***.sectioncompositedistribution**
Composites *DISTRIBUTION TABLE definition to be included on *ASSEMBLY level

***.shellsectioncomposite**
Composites *SHELL SECTION definition to be included on *PART level

Table 4.1: Output files generated from MpCCI Mapper

4.2 ANSYS Mechanical APDL

4.2.1 File Format

The ANSYS Mechanical APDL model must have common database (*.cdb) format to import into MpCCI Mapper . Only the keywords ”NBLOCK”,”EBLOCK”,”CMBLOCK”,”ET” and ”KEYOP” are currently supported.

4.2.2 Element Types

Shells

Full integrated trians (3 nodes) and quads(4 nodes) SHELL181.

Solids

Tetrahedrons (4 & 10 nodes), hexahedrons (8 & 20 nodes), pyramid (5 nodes) and prisms (6 nodes) are supported for reading and writing. Detailed list of supported ANSYS elements SOLID5, SOLID45, SOLID46, SOLID62, SOLID64, SOLID65, SOLID69, SOLID70, SOLID87, SOLID92, SOLID96, SOLID97, SOLID98, SOLID123, SOLID127, SOLID148, SOLID164, SOLID168, SOLID185, SOLID186, SOLID187, SOLID227, SOLID232, SOLID237 and SOLID285.

4.2.3 Transferable Quantities

Quantities THICKNESS, TEMPERATURE, STRESS, STRAIN, PLASTIC_STRAIN, PRESSURE and ORIENTATION_TENSOR are supported for mapping and writing. Boudary condition type INISTATE use defaut coordinate system identifier 0.

Quantity	Boundary Condition Type
THICKNESS	`Tabular $NodeId $Value`
TEMPERATURE	`BF, ,TEMP`
STRESS	`INISTATE, SET, DTYP, S`
STRAIN	`INISTATE, SET, DTYP, EPEL`
PLASTIC_STRAIN	`INISTATE, SET, DTYP, EPPL`
PRESSURE	`SFE,,,PRES`
ORIENTATION_TENSOR	Moldflow xml format, Digimat *.dof
WELD_LINE	Element set or Digimat *.dof
DISPLACEMENTS	`nblock` coordinate with applied displacement

4.3 AutoForm R4 converter

MpCCI Mapper command to convert exported CSV format reads as follows

```
> mapper.exe -csv2dyna FILE_1 FILE_2 FILE_3 FILE_4
```

where each file in argument list needs to contain parameters as specifyed in table 4.2.

File	Description
FILE_1	*_NodeData_Thickness containing nodal coordinates and ids
FILE_2	*_ElementData_Thickness containing element topology and thickness
FILE_3	*_ElementData_Plastic_Strain containing element topology plastic strain
FILE_4	Result_File_Name

Table 4.2: AutoForm converter parameter list

Required CSV format per output file:

1. *_NodeData_Thickness:

   Node Idx,XCoord,YCoord,ZCoord,,Info

2. *_ElementData_Thickness:

   Element Idx,Node Idx 1,Node Idx 2,Node Idx 3,Thickness,,Info

3. *_ElementData_Plastic_Strain (5 points over thickness):

   Element Idx,Node Idx 1,Node Idx 2,Node Idx 3,Plastic Strain Layer -2,Plastic Strain Layer -1,Plastic Strain,Plastic Strain Layer 1,Plastic Strain Layer 2,,Info

4.4 Cadmould

4.4.1 File Format

Cadmould geometry file *.cfe is supported. The file contains the element definitions, the node coordinates, the number of nodes and elements and an identifier if it is a surface mesh or a midplane mesh. MpCCI Mapper can import both midplane and surface meshes. Surface meshes consist of pairs of triangles which are internally converted into a layered prism mesh. Midplane meshes consist of triangular shell elements. As result format binary *.car files containing ORIENTATION_TENSOR information can be read.

4.4.2 Transferable Quantities

Quantity	Read	Write
THICKNESS	x	-
ORIENTATION_TENSOR	x	-

4.5 Indeed

4.5.1 File Format

For code Indeed reading of ASCII result file (*.asc) and geometry file (*.inc) is supported.

4.5.2 Element Types

The 6- and 8-node Crisfield-Solid-Shell elements are supported for reading.

4.5.3 Transferable Quantities

Quantity	Read	Write
THICKNESS	x	-
STRESS	x	-
PLASTIC_STRAIN	x	-

4.6 MSC Simufact

4.6.1 File Format

All MSC.Simufact products, Additive, Forming and Welding, offer a configurable result export utilizing the Universal File format (*.unv), which is fully supported by MpCCI Mapper .

4.6.2 Element Types

MpCCI Mapper supports reading of all element types available in MSC Simufact product. The interface comprises reading solid, thick shell and shell elements.

4.6.3 Transferable Quantities

Both, nodal and element integration point results can be read from MSC.Simufact Universal files. The data in imported in SI units. The following list of output variables is automatically handeled as MpCCI Mapper quantitites:

Quantity	Read	Write
THICKNESS	x	-
STRESS	x	-
PLASTIC_STRAIN	x	-
TEMPERATURE	x	-
DISPLACEMENTS	x	-

4.7 Forge

4.7.1 File Format

The Universal File format (*.unv), which can be exported via the IDEAS export interface in GLPre, is supported by MpCCI Mapper .

4.7.2 Element Types

Shells

Full integrated trians (3 & 6 nodes) and quads(4 nodes) are supported for reading.

Solids

Tetrahedrons (4 & 10 nodes), hexahedrons (8 & 20 nodes) and prisms (6 nodes) are supported for reading.

4.7.3 Transferable Quantities

In general nodal based (card format 55), element based (card format 56) and general card format 2414 quantities (nodal or element based) can be read from Universal File format. Variables TEMPERATURE and STRESSTENSOR are automatically detected and assigned as MpCCI Mapper quantities TEMPERATURE and STRESS.

4.8 LSDyna

4.8.1 File Format

LSDyna keyword format (*.k,*.key,*.dyn,*dynain) is supported for reading and writing.

4.8.2 Element Types

*ELEMENT_SHELL Card

Full integrated trians (3 nodes) and quads(4 nodes). Arbitrary out of plane integration types up to 16 integration points out of plane.

*ELEMENT_SOLID Card

Tetrahedrons (4 & 10 nodes), hexahedrons (8 nodes) and prisms (6 nodes) are supported for reading and writing. Element formulations -2, -1, 1, 2, 10, 13, 15, 16 and 115 form solid section can be used for quantity generation.

4.8.3 Transferable Quantities

Quantity	Read	Write	Boundary Condition or Element Card
THICKNESS	x	x	`*ELEMENT_SHELL_THICKNESS`
STRESS	x	x	`*INITIAL_STRESS_SHELL`
	x	x	`*INITIAL_STRESS_SOLID`
STRAIN	x	x	`*INITIAL_STRAIN_SHELL`
	x	x	`*INITIAL_STRAIN_SOLID`
PLASTIC_STRAIN	x	x	`*INITIAL_STRESS_SHELL`
	x	x	`*INITIAL_STRESS_SOLID`
HISTORY Variables	x	x	`*INITIAL_STRESS_SHELL`
	x	x	`*INITIAL_STRESS_SOLID`
ORIENTATION_TENSOR	-	x	`*INITIAL_STRESS_SHELL`
	x	x	`*ELEMENT_SOLID_ORTHO`
	-	x	`*INITIAL_STRESS_SOLID`
ELEMENT_SHELL_BETA	x	x	`*ELEMENT_SHELL_THICKNESS`
TEMPERATURE	-	x	`*INITIAL_TEMPERATURE_NODE`

Table 4.3: MpCCI Mapper quantities in LSDyna solver format

4.8.4 Composite Material Model Features

For LSDyna Version R 10 MpCCI Mapper supports direct mapping of locally orthotropic material axes from *MAT_REINFORCED_THERMOPLASTIC (*MAT_249) to *MAT_LAMINATED_COMPOSITE_FABRIC (*MAT_58). Therefor direction of fibers, stored in certain HISTORY Variables, are gathered in separated vector quantites:

Quantity Name	HISTORY Variable	Comment
PLY_VECTOR_1	POSTV dependent	Direction 1st fiber (global coordinates)
PLY_VECTOR_2	POSTV dependent	Direction 2nd fiber (global coordinates)

Table 4.4: MpCCI Mapper quantities for composites.

4.8.5 Injection Molding Features

The fiber ORIENTATION_TENSOR result shows the probability of fiber alignment in the specified principal direction at the end of a injection molding process. Depending on the degree of orientation MpCCI Mapper can classify and export 15 different placeholder material cards as well as section solid assignments for solid elements. For each solid element local material axes are defined using *ELEMENT_SOLID_ORTHO card. For material #1 a *MAT_ELASTIC template and for material 2-15 a *MAT_ANISOTROPIC_ELASTIC template is written to output files.

$$\begin{pmatrix} a_{11} = 0.33 \\ a_{22} = 0.33 \\ a_{33} = 0.33 \end{pmatrix} \qquad \begin{pmatrix} a_{11} = 0.70 \\ a_{22} = 0.20 \\ a_{33} = 0.10 \end{pmatrix} \begin{pmatrix} a_{11} = 0.70 \\ a_{22} = 0.30 \\ a_{33} = 0.00 \end{pmatrix}$$

$$\begin{pmatrix} a_{11} = 0.40 \\ a_{22} = 0.30 \\ a_{33} = 0.30 \end{pmatrix} \begin{pmatrix} a_{11} = 0.40 \\ a_{22} = 0.40 \\ a_{33} = 0.20 \end{pmatrix} \qquad \begin{pmatrix} a_{11} = 0.80 \\ a_{22} = 0.10 \\ a_{33} = 0.10 \end{pmatrix} \begin{pmatrix} a_{11} = 0.80 \\ a_{22} = 0.20 \\ a_{33} = 0.00 \end{pmatrix}$$

$$\begin{pmatrix} a_{11} = 0.50 \\ a_{22} = 0.30 \\ a_{33} = 0.20 \end{pmatrix} \begin{pmatrix} a_{11} = 0.50 \\ a_{22} = 0.40 \\ a_{33} = 0.10 \end{pmatrix} \begin{pmatrix} a_{11} = 0.50 \\ a_{22} = 0.50 \\ a_{33} = 0.00 \end{pmatrix} \qquad \begin{pmatrix} a_{11} = 0.90 \\ a_{22} = 0.10 \\ a_{33} = 0.00 \end{pmatrix}$$

$$\begin{pmatrix} a_{11} = 0.60 \\ a_{22} = 0.20 \\ a_{33} = 0.20 \end{pmatrix} \begin{pmatrix} a_{11} = 0.60 \\ a_{22} = 0.30 \\ a_{33} = 0.10 \end{pmatrix} \begin{pmatrix} a_{11} = 0.60 \\ a_{22} = 0.40 \\ a_{33} = 0.00 \end{pmatrix} \qquad \begin{pmatrix} a_{11} = 1.00 \\ a_{22} = 0.00 \\ a_{33} = 0.00 \end{pmatrix}$$

Figure 4.1: Fiber orientation tensor classes for element solid ortho definition.

Id	Material Title	SECTION_SOLID_TITLE	Comment
1	MATERIAL_ISOTROPIC	PART_ISOTROPIC	a11 = a22 = a33 = 0.33
2	MATERIAL_0_4_0_3_0_3	PART_0_4_0_3_0_3	a11 = 0.4, a22 = 0.3, a33 = 0.3
3	MATERIAL_0_4_0_4_0_2	PART_0_4_0_4_0_2	a11 = 0.4, a22 = 0.4, a33 = 0.3
4	MATERIAL_0_5_0_3_0_2	PART_0_5_0_3_0_2	a11 = 0.5, a22 = 0.3, a33 = 0.2
5	MATERIAL_0_5_0_4_0_1	PART_0_5_0_4_0_1	a11 = 0.5, a22 = 0.4, a33 = 0.1
6	MATERIAL_0_5_0_5_0_0	PART_0_5_0_5_0_0	a11 = 0.5, a22 = 0.5, a33 = 0.0
7	MATERIAL_0_6_0_2_0_2	PART_0_6_0_2_0_2	a11 = 0.6, a22 = 0.2, a33 = 0.2
8	MATERIAL_0_6_0_3_0_1	PART_0_6_0_3_0_1	a11 = 0.6, a22 = 0.3, a33 = 0.1
9	MATERIAL_0_6_0_4_0_0	PART_0_6_0_4_0_0	a11 = 0.6, a22 = 0.4, a33 = 0.0
10	MATERIAL_0_7_0_2_0_1	PART_0_7_0_2_0_1	a11 = 0.7, a22 = 0.2, a33 = 0.1
11	MATERIAL_0_7_0_3_0_0	PART_0_7_0_3_0_0	a11 = 0.7, a22 = 0.3, a33 = 0.0
12	MATERIAL_0_8_0_1_0_1	PART_0_8_0_1_0_1	a11 = 0.8, a22 = 0.1, a33 = 0.1
13	MATERIAL_0_8_0_2_0_0	PART_0_8_0_2_0_0	a11 = 0.8, a22 = 0.2, a33 = 0.0
14	MATERIAL_0_9_0_1_0_0	PART_0_9_0_1_0_0	a11 = 0.9, a22 = 0.1, a33 = 0.0
15	MATERIAL_1_0_0_0_0_0	PART_1_0_0_0_0_0	a11 = 1.0, a22 = 0.0, a33 = 0.0

Table 4.5: *ELEMENT_SOLID_ORTHO output generated by MpCCI Mapper from a mapped ORIENTATION_TENSOR.

$$\begin{pmatrix}0.50&0&0\\0&0.50&0\\0&0&0\end{pmatrix}\begin{pmatrix}0.55&0&0\\0&0.45&0\\0&0&0\end{pmatrix}\begin{pmatrix}0.60&0&0\\0&0.40&0\\0&0&0\end{pmatrix}\begin{pmatrix}0.65&0&0\\0&0.35&0\\0&0&0\end{pmatrix}\begin{pmatrix}0.70&0&0\\0&0.30&0\\0&0&0\end{pmatrix}\begin{pmatrix}0.75&0&0\\0&0.25&0\\0&0&0\end{pmatrix}\begin{pmatrix}0.80&0&0\\0&0.20&0\\0&0&0\end{pmatrix}$$

$$\begin{pmatrix}0.85&0&0\\0&0.15&0\\0&0&0\end{pmatrix}\begin{pmatrix}0.90&0&0\\0&0.10&0\\0&0&0\end{pmatrix}\begin{pmatrix}0.95&0&0\\0&0.05&0\\0&0&0\end{pmatrix}\quad\begin{pmatrix}1.0&0&0\\0&0&0\\0&0&0\end{pmatrix}$$

Figure 4.2: Fiber orientation tensor classes for element shell composite definition.

Id	Material Title	Comment
101	MATERIAL_0_50_0_50_ISOTROPIC	a11 = a22 = 0.50
102	MATERIAL_0_55_0_45	a11 = 0.55, a22 = 0.45
103	MATERIAL_0_60_0_40	a11 = 0.60, a22 = 0.40
104	MATERIAL_0_65_0_35	a11 = 0.65, a22 = 0.35
105	MATERIAL_0_70_0_30	a11 = 0.70, a22 = 0.30
106	MATERIAL_0_75_0_25	a11 = 0.75, a22 = 0.25
107	MATERIAL_0_80_0_20	a11 = 0.80, a22 = 0.20
108	MATERIAL_0_85_0_15	a11 = 0.85, a22 = 0.15
109	MATERIAL_0_90_0_10	a11 = 0.90, a22 = 0.10
110	MATERIAL_0_95_0_05	a11 = 0.95, a22 = 0.05
111	MATERIAL_1_00_0_00	a11 = 1.00, a22 = 0.00

Table 4.6: *ELEMENT_SHELL_COMPOSITE output generated by MpCCI Mapper from a mapped ORIENTATION_TENSOR.

*.key	File with model *NODE definition Include statements for *.key.thick and *.key.stressstrain
*.key.thick	Model *ELEMENT definition with THICKNESS
*.key.stressstrain	STRESS, STRAIN, PLASTIC_STRAIN and HISTORY Variables
*_complete.key	Combinend *.key,*.key.thick and *.key.stressstrain output
*_with_displacements.key	Optional file containing morphed coordinates when mapping solid to shell

Table 4.7: Output files generated from MpCCI Mapper

4.8.6 Output Files

Historyvariables

MpCCI Mapper allows a redefinition of the output sequence of history variables via ASCII-based configuration file over the specific environment variable or via the file writer settings dialog 3.1.2. Using the environment variable option, a user has to specify the variable

```
LSDYNA_HISTORY_CONFIGURATION_FILE=Global_Path_To_File/File
```

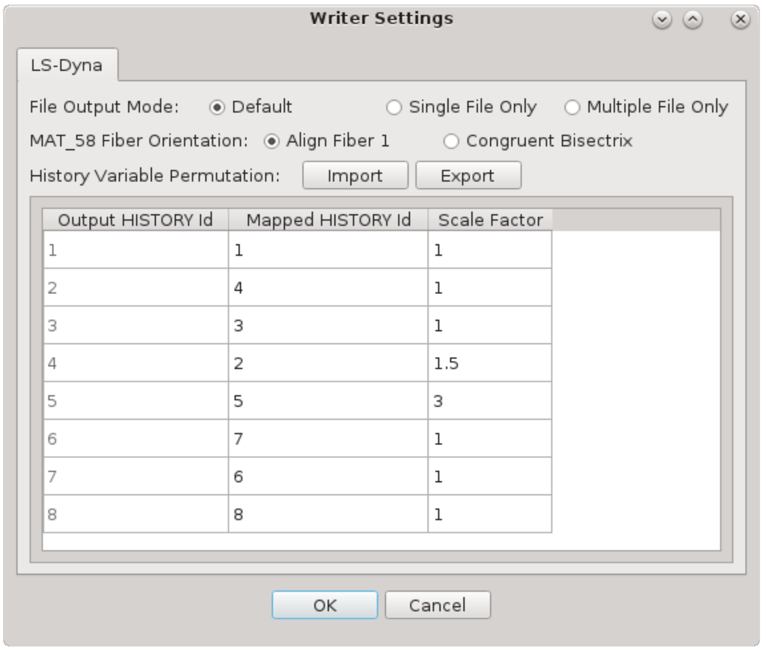
which defines the global path to the configuration file which shall be used for history variable exchange. Starting which keyword “MapperHistoryVariableConfiguration” the configuration is given in tabular format

```
MapperHistoryVariableConfiguration
OriginalHistoryId NewHistoryId  ScalingFactor
```

In some cases a mapping of history variables between different material models require an additional scaling of a variable. To scale a variable on result export an auxiliary scaling factor can be specified in column number three. As default a scaling factor of 1.0 is applied if the column entry is not specified.
Example:

```
MapperHistoryVariableConfiguration
1 1
2 4 1.5
3 3
4 2
5 5 3.0
6 7
7 6
8 8
```

Figure 4.4 shows the history variable handling when writing a LSDyna model. Here Variable 2 and 4 as well as 6 and 7 are swapped in output order. Variable 2 is additionally scaled by a factor of 1.5, variable 5 will be scaled by a factor of 3.0.



File Output Mode: ● Default ○ Single File Only ○ Multiple File Only

MAT_58 Fiber Orientation: ● Align Fiber 1 ○ Congruent Bisectrix

History Variable Permutation: Import Export

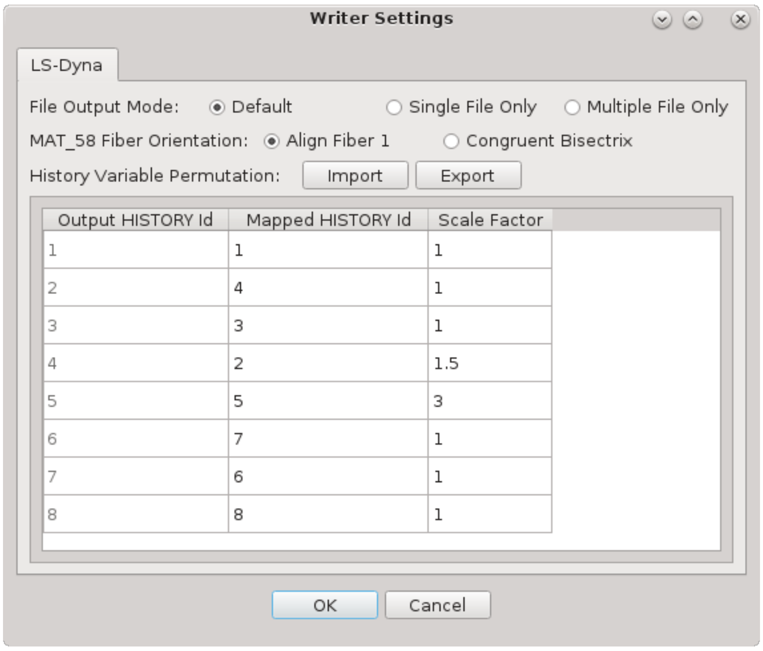
Output HISTORY Id	Mapped HISTORY Id	Scale Factor
1	1	1
2	4	1
3	3	1
4	2	1.5
5	5	3
6	7	1
7	6	1
8	8	1

Figure 4.3: History variable permutation imported in writer settings dialog.

The same output can be generated using short notation

```
MapperHistoryVariableConfiguration
2 4 1.5
4 2
5 5 3.0
6 7
7 6
```

If the new history id is set to “0”, the variable is set to zero for all integration points and thus treated as empty.



Writer Settings

LS-Dyna

File Output Mode: ● Default ○ Single File Only ○ Multiple File Only

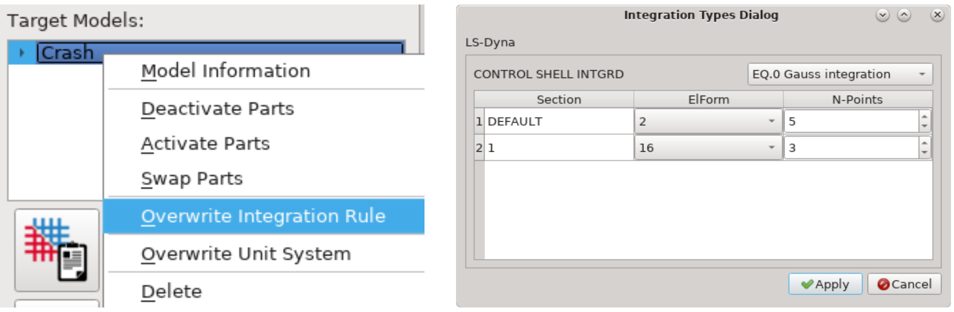
MAT_58 Fiber Orientation: ● Align Fiber 1 ○ Congruent Bisectrix

History Variable Permutation: Import Export

Output HISTORY Id	Mapped HISTORY Id	Scale Factor
1	1	1
2	4	1
3	3	1
4	2	1.5
5	5	3
6	7	1
7	6	1
8	8	1

Figure 4.4: History variable permutation imported in writer settings dialog.

Element formulation and integration scheme



Integration Types Dialog

LS-Dyna

CONTROL SHELL INTGRD: EQ.0 Gauss integration

	Section	ElForm	N-Points
1	DEFAULT	2	5
2	1	16	3

Figure 4.5: Redefinition of Elform and through thickness integration scheme

If the LSDyna model is used as target model, MpCCI Mapper supports a check and redefinition of shell element formulation and through thickness integration rule (and number of points though thickness) for each shell section within the model. Using right-click option on target model name in “Target Models” tree 3.4.1, then selecting “Overwrite Integration Rule” an integration types dialog opens 4.5. Here, a default shell section integration formula is given as well for each shell section within the model an additional data row to change element fomulation (reduced of fully integrated) as well as the number of integration points though thickness.

Note: If element formulation of the number of integration points though thickness is changed, all defined quantities of the model will be deleted. To obtain data on specified element formulation / number of integration points please re-run mapping step.

4.9 Marc / COPRA FEA

4.9.1 File Format

MpCCI Mapper supports reading of ASCII based MSC Marc result format (*.t19). By default MSC Marc does export binary .t16 files only which are not supported. To request an ASCII .t19 select the relevant Job in MSC Mentat Preprocessor, and go 'Job Properties' → 'Job Results' → 'Post File' and activate 'Binary & ASCII' option. Then, both ASCII and Binary file will be written during simulation.

4.9.2 Element Types

Mapper Element Type	MSC Marc Element Types
SHELL QUAD 4	3,11,18,22,39,75,85,114,115,139,140,198
SHELL TRIAN 3	6,37,50,138,158,196,210
SHELL QUAD 8	30,32,41,72,86
SHELL TRIAN 6	49,124,125
SOLID TETRA 4	134,135,157
SOLID TETRA 10	127,130,133,184
SOLID HEXA 8	7,43,109,113,117,120,123
SOLID HEXA 20	21,44
SOLID WEDGE 6	136,137,204

Table 4.8: Supported MSC Marc element types

4.9.3 Transferable Quantities

MpCCI Mapper supports reading of nodal displacements and computes deformed geometry for initial definition of nodal coordinates.

Quantity	Read	Write
TEMPERATURE	x	-
STRESS	x	-
STRAIN	x	-
PLASTIC_STRAIN	x	-

4.10 Moldex3D

4.10.1 File Format

Moldex geometry file exported to Abaqus *.inp is supported for reading. Each variable has to be written to separated file as described in transferable quantitites section.
REMARK: WELD_LINE variable will be clipped to 1.

4.10.2 Transferable Quantities

Quantity	File extension	Read	Write
THICKNESS	-	x	-
ORIENTATION_TENSOR	.o2d	x	-
WELD_LINE	.w2d	x	-
STRESS	.s2d	x	-
TEMPERATURE	.t2d	x	-

4.11 Moldflow

4.11.1 File Format

Moldflow geometry file (*.udm) and exported to PATRAN neutral (*.pat) is supported. The file contains the element definitions and the node coordinates. ORIENTATION_TENSOR results as generic .xml, PATRAN neutral (*.nod) or *.ele* files - for orientation tensor on midplane meshes - are supported for reading.

4.11.2 Transferable Quantities

Quantity	Read	Write
THICKNESS	x	-
ORIENTATION_TENSOR	x	-
WELD_LINE	x	-

4.12 Nastran

4.12.1 File Format

MpCCI Mapper supports reading of ASCII Nastran Bulk data format (*.nas,*.bdf). Nodal coordinates can be either in long (GRID*) or short (GRID) format. Nastran so called free format is not supported.

4.12.2 Element Types

Triangles `CTRIA3` and quads `CQUAD4` are supported for reading and writing.

4.12.3 Transferable Quantities

Quantity	Read	Write
THICKNESS	x	x
STRESS	-	x
PLASTIC_STRAIN	-	x
PRESSURE	x	x

Quantity	Boundary Condition Type
THICKNESS	element description card format
STRESS	ISTRSSH for SOL 700 only
PLASTIC_STRAIN	ISTRSSH for SOL 700 only
PRESSURE	`PLOAD2`

4.13 OptiStruct

4.13.1 File Format

MpCCI Mapper supports reading of ASCII OptiStruct geometry files *.fem. Tiangular and quad shell elements with thickness information can be read. As solid elements four node tetra, six node penta and eight node hexa are supported.

4.13.2 Transferable Quantities

Quantity	Read	Write	Card format
THICKNESS	x	x	CTRIA, CQUAD
STRESS	-	x	INISTRS
TEMPERATURE	-	x	TEMP
DISPLACEMENTS	-	x	GRID

4.14 PAM

4.14.1 File Format

MpCCI Mapper supports the following native PAM solver ASCII formats:

PAMStamp mapping files	(*M00-*M99)
PAMCrash input deck	(*.pc,*.ps)

4.14.2 Element Types

Full integrated triangles (3 nodes) and quads(4 nodes) with arbitrary out of plane integration number can be read. Only active parts are written into *M00-*M99 format for both native solver input types.

4.14.3 Transferable Quantities

Quantity	Read	Write	Boundary Condition
THICKNESS	x	x	THIC
STRESS	x	x	STRS
STRAIN	x	x	STRN
PLASTIC_STRAIN	x	x	PLAS
PRESSURE	-	x	PREFA
HV__	x	x	HV__
ORIENTATION_TENSOR	-	x	ORTH

4.15 Radioss

4.15.1 File Format

Radioss Result	(*Yxxx)	Reading and writing is supported
Radioss Result	(*.sta)	Reading and writing is supported
Radioss Input	(*D00, .rad)	Reading is supported

4.15.2 Element Types

Fully integrated and reduced integrated triangles and quads are supported.

4.15.3 Transferable Quantities

Quantity	Read	Write
THICKNESS	x	x
STRESS	x	x
PLASTIC_STRAIN	x	x

4.16 Sysweld

4.16.1 File Format

`*.asc`	Model information file
`*_trans.asc`	Mechanical or thermal result file Stresses, strain, plastic strain and young modul is extracted. The effectiv plastic strain is extracted from plastic strain.
`*_phase.asc`	Defining names for the phases.

The syntax is that of the sysweld files and looks like this

`SYSWELD_PHASE_DEFINITION`	First line of file is the magic word.
`BEGIN_PHASEDEF`	Starting to define the mnemonics
`1 "MARTENSIT"`	Name of first phase
`2 "AUSTENIT"`	Name of second phase
`5 "SOMEOTHERUSERDEFINEDPHASE"`	Name of phase five
`END_PHASEDEF`	End of definition statments

Output The files exept the `*_phase.asc` from above are written or are patched where values have changed.

!! Attention, currently the writer is designed to write the mapped values of a metal forming result. To keep equlibrium the elastic strains are calculated from the residual stresses. As only plastic strains can be written and as the total strain needs to be 0 (displacements are assumed to be 0) the plastic strain is set to the calculated negative strain values. The young modulus is used to calculate the strains.

5 Other Formats

5.1 Argus / Aramis

Argus and Aramis offer a configurable output format which can be imported in MpCCI Mapper (`*.txt`). Thickness information and strain can be imported.

5.1.1 File Format

To define an initial thickness (here 1.00 mm) use following string in file header comment block:

```
# Ausgangsblechdicke: 1.00mm
```

1	2	3	4	5	6	7	8	9	10
s	u	v	`x_coord`	`y_coord`	`z_coord`	`phi_1`	`phi_2`	`phi_3`	%thickness reduction

Table 5.1: Order of Variables required for MpCCI Mapper

5.2 Atos

5.2.1 File Format

The 'Gesellschaft für Optische Messtechnik' offers an export plugin that allows geometry and thickness information export in polygon file format (`*.ply`) which is supported for reading in MpCCI Mapper . This plugin can be received directly from GOM `http://www.gom.com`.

5.3 AutoGrid

5.3.1 File Format

MpCCI Mapper supports AutoGrid `*.dat` file import for software version 4.1.x.x and 4.2.x.x. Thickness and strain can be imported.

5.4 STL

Eighter ASCII or binary STL-files can be imported in MpCCI Mapper using the common `*.stl` file extension.

5.5 VTK

ASCII based VTK legacy files (*.vtk) can be imported in MpCCI Mapper .

5.5.1 Element Types

VTK	MpCCI Mapper	Comment
VTK_TRIANGLE	TRIAN	3-node shell element
VTK_QUAD	QUAD	4-node shell element
VTK_QUADRATIC_TRIANGLE	TRIAN6	6-node shell element
VTK_QUADRATIC_QUAD	QUAD8	8-node shell element
VTK_TETRA	TETRA	4-node volume element
VTK_HEXAHEDRON	HEXA	8-node volume element
VTK_WEDGE	WEDGE	6-node volume element
VTK_QUADRATIC_TETRA	TETRA10	10-node volume element
VTK_QUADRATIC_HEXAHEDRON	HEXA20	20-node volume element

Table 5.2: Translation of VTK elements in MpCCI Mapper

6 Thirdparty License Information

6.1 Boost

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:
The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

6.2 LibQxt

Copyright (c) 2006 - 2011, the LibQxt project. See the Qxt AUTHORS file for a list of authors and copyright holders. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the LibQxt project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO

EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
http://libqxt.org foundation@libqxt.org

6.3 Libtiff

Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.
Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that (i) the above copyright notices and this permission notice appear in all copies of the software and related documentation, and (ii) the names of Sam Leffler and Silicon Graphics may not be used in any advertising or publicity relating to the software without the specific, prior written permission of Sam Leffler and Silicon Graphics.

THE SOFTWARE IS PROVIDED ”AS-IS” AND WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

6.4 Libstdc++

Copyright (C) 2008-2020 FSF <https://www.fsf.org>

6.5 Libwebp

Copyright (c) 2010, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- Neither the name of Google nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS ”AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

6.6 LibXBC

Copyright (c) 2000 Keith Packard
Copyright (c) 2006 Jamey Sharp
Copyright (c) 2007-2008 Vincent Torri vtorri@univ-evry.fr
Copyright (c) 2007 Bart Massey
Copyright (c) 2008-2009 Julien Danjou julien@danjou.info
Copyright (c) 2008 Arnaud Fontaine arnau@debian.org
Copyright (c) 2008 Bart Massey bart@cs.pdx.edu
Copyright (c) 2008 Ian Osgood iano@quirkster.com
Copyright (c) 2008 Jamey Sharp jamey@minilop.net
Copyright (c) 2008 Josh Triplett josh@freedesktop.org
Copyright (c) 2008 Ulrich Eckhardt doomster@knuut.de

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the ”Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ”AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the names of the authors or their institutions shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the authors.

6.7 pugixml

This software is based on pugixml library (https://pugixml.org).
pugixml is Copyright (C) 2006-2019 Arseny Kapoulkine.

6.8 HDF5

Copyright Notice and License Terms for HDF5 (Hierarchical Data Format 5) Software Library and Utilities

HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright (c) 2006-2018, The HDF Group.

NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities Copyright (c) 1998-2006, The Board of Trustees of the University of Illinois.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted for any purpose (including commercial purposes) provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or materials provided with the distribution.

3. Neither the name of The HDF Group, the name of the University, nor the name of any Contributor may be used to endorse or promote products derived from this software without specific prior written permission from The HDF Group, the University, or the Contributor, respectively.

DISCLAIMER:
THIS SOFTWARE IS PROVIDED BY THE HDF GROUP AND THE CONTRIBUTORS ”AS IS” WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED.IN NO EVENT SHALL THE HDF GROUP OR THE CONTRIBUTORS BE LIABLE FOR ANY DAMAGES SUFFERED BY THE USERS ARISING OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You are under no obligation whatsoever to provide any bug fixes, patches, or upgrades to the features, functionality or performance of the source code (”Enhancements”) to anyone; however, if you choose to make your Enhancements available either publicly, or directly to The HDF Group, without imposing a separate written license agreement for such Enhancements, then you hereby grant the following license: a non-exclusive, royalty-free perpetual license to install, use, modify, prepare derivative works, incorporate into other computer software, distribute, and sublicense such enhancements or derivative works thereof, in binary and source code form.

Limited portions of HDF5 were developed by Lawrence Berkeley National Laboratory (LBNL). LBNL's Copyright Notice and Licensing Terms can be found here: COPYING_LBNL_HDF5 file in this directory or at
http://support.hdfgroup.org/ftp/HDF5/releases/COPYING_LBNL_HDF5.

Contributors: National Center for Supercomputing Applications (NCSA) at the University of Illinois, Fortner Software, Unidata Program Center (netCDF), The Independent JPEG Group (JPEG), Jean-loup Gailly and Mark Adler (gzip), and Digital Equipment Corporation (DEC).

Portions of HDF5 were developed with support from the Lawrence Berkeley National Laboratory (LBNL) and the United States Department of Energy under Prime Contract No. DE-AC02-05CH11231.

Portions of HDF5 were developed with support from the University of California, Lawrence Livermore National Laboratory (UC LLNL). The following statement applies to those portions of the product and must be retained in any redistribution of source code, binaries, documentation, and/or accompanying materials:

This work was partially produced at the University of California, Lawrence Livermore National Laboratory (UC LLNL) under contract no. W-7405-ENG-48 (Contract 48) between the U.S. Department of Energy (DOE) and The Regents of the University of California (University) for the operation of UC LLNL.

DISCLAIMER:
This work was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately-owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

6.9 PCRE Library

Copyright (c) 1997-2016 University of Cambridge
Copyright (c) 2009-2016 Zoltan Herczeg
Copyright (c) 2007-2012 Google Inc.
Copyright (c) 2013-2013 Tilera Corporation (jiwang@tilera.com)
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of

6.10 VTK