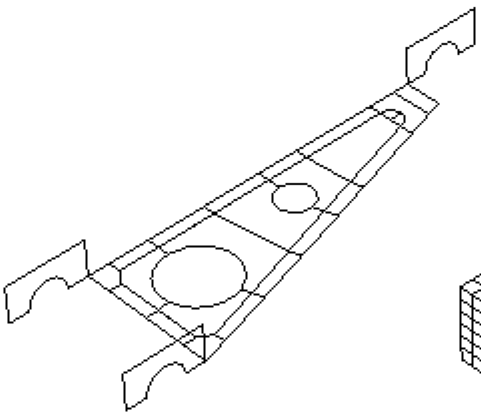
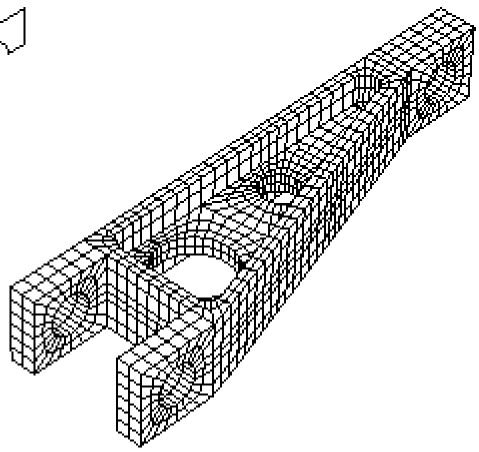
MAKROS

Pre- und Post processing of Finite Elements AutoCAD-extension for the creation of finite element models





Demo-Installation: ftp://ftp.bauv.unibw-muenchen.de/pub/Makros-A/

Prof. Dr.-Ing. Günther Böge
Universität der Bundeswehr München
D-85577 Neubiberg
guenther.boege@unibw-muenchen.de
Tel: +49 89 6004 3404
Fax: +49 89 6004 4136

Introduction

Different Program Versions

There are three different versions of MAKROS:

Makrose.exe/Makros.exe: Independent application (English/German),
Makrosae.exe/Makrosa.exe: ADS application of AutoCAD (English/German),
Makrosarxe.arx/Makrosarx.arx: ARX application of AutoCAD (English/German).

Using makrosarx.exe, data transfer between AutoCAD and MAKROS is much faster.

Functionality

MAKROS is a program for Finite Element Pre- and Postprozessing. It may be started as independent application or as an ARX respectively ADS application to AutoCAD, where it is possible to exchange data between AutoCAD and MAKROS.

MAKROS offers the following functions:

Creation of Finite Element (FE) models

The creation is done in following steps:

First, a geometry model is created within AutoCAD or with MAKROS commands. This model uses as few as possible macro elements to approximate the geometry of the given structure. Within these terms „macro element's" are large elements whose edges may also be spline curves. All macro elements give the "Macro model" of the structure.

The macro model is transferred to MAKROS if it is created in AutoCAD.

MAKROS specifies the parameters for the subdivision of the macro elements into finite elements.

MAKROS calculates the subdivision of the macro model and creates the finite element model, which may be transferred to AutoCAD.

With further commands of MAKROS this finite element mesh can be logically checked and possibly geometrically modified within AutoCAD or with MAKROS commands.

Additional commands of MAKROS can be used to assign constraints, loads, element properties and material data.

The resulting FE model can be saved in a text file using the neutral PATRAN, the NASTRAN or a special interface specification.

Using command "Load DLL" it is also possible to invoke own functions for the generation of nodes and elements.

Interface between AutoCAD and other processing tools

MAKROS can also be used to translate geometry data from AutoCAD to other processing tools. AutoCAD data can be read directly from an AutoCAD session or from a DXF-file. AutoCAD entities are translated to line, surface or solid elements. Geometrical data can be checked within MAKROS and possibly be modified by several functions and made available for further processing within simply structured text files. MAKROS can read most entity types from AutoCAD. Blocks and polylines composed of line and curved segments prior must be broken up into basic components by the AutoCAD command „xplode".

Post processing

MAKROS offers functions for visualization of calculation results received from other FE programs.

Invoking MAKROSARXE, MAKROSAE, MAKROSE

Makrosarxe is loaded as a DLL with the AutoCAD command ARX/Load, the file "makrosarxe.arx" must then be selected in a file selection dialog. The DLL can not be unloaded using ARX/Unload, it remains loaded until AutoCAD is finished. Makrosarxe may also be loaded using the AutoCAD command (arxload "makrosarxe").

Makrosae is activated with the AutoCAD command (xload "makrosae"). Please note the parentheses which must surround AutoLISP command "xload". Makrosae can be unloaded with (xunload "makrosae").

<u>Note</u>: Starting Makrosae or Makrosarxe within AutoCAD using "xload" respectively "arxload", AutoCAD needs the location of the executables and menu definitions of MAKROS. The directory of these files must be added to the support path of AutoCAD (see AutoCAD menu Options/Preferences/Environment/Support).

Makrose.exe can be started as a stand alone application, also it can be associated with several file types and so it can be easily started by double clicking the relevant file name. Supported file types are:

*.dem: Demo files for automated run,

*.mes: Binary macro element structure is loaded and displayed,

*.fes: Binary finite element model is loaded and displayed.

Commands, switching control between AutoCAD and MAKROS

The invocation of commands within MAKROS is done by selecting menus on the menu bar of MAKROS.

Each command consists of a single keyword only. After command invocation an associated dialog box pops up where additional parameters can be specified. By pressing button "Help" within a dialog box, the user can become detailed information about the command.

Execution of the command is started by pressing button "OK", pressing button "Cancel" will leave the command without execution.

Messages of MAKROS are displayed within the protocol window of MAKROS. Popup windows are brought up if yes or no answers of the user are required.

If MAKROS (makrosae or makrosarxe) is loaded from within AutoCAD, the control of the process is either within AutoCAD or within MAKROS and only commands given in the currently active program are executed. With the menu commands "MAKROSA" ("m_makrosa" in the command line of AutoCAD) respectively "AutoCAD" can be switched between both programs. Giving the command "MAKROSA" ("m_makrosa") within AutoCAD, the menu bar within the MAKROS window will be shown. Commands have then to be selected only within this menu. All commands given within AutoCAD are delayed. After invoking the command "AutoCAD" within the menu of MAKROS, this menu bar will be hidden and following commands only can be given within AutoCAD.

Loading makrosae from AutoCAD, the menu bar of AutoCAD is automatically expanded by the menu group „MAKROSA". Loading makrosarxe the command "m_makrosa" must first be given in the command line of AutoCAD, then control is transferred to MAKROS and the dialog box of the AutoCAD command "menuload" pops up, where the AutoCAD menu file "makrosa.mnc" must be loaded and the menu group "MAKROSA" added to the menu bar of AutoCAD.

The menu group "MAKROSA" has only the menu point "MAKROSA" that invokes the command "m_makrosa" to transfers control to MAKROS.

Programming interface

The functionality of MAKROS may be expanded by own functions. These functions must be provided in one or more DLLs. The commands **DLL-Function** (see chapter "Commands to generate new nodes and elements")

and **Interface (DLL)** (see chapter "Interfaces to FE-Programs") allow to call functions from DLLs. Folder "makrosa/dll" contains some examples for the development of DLL functions.

Graphical output, OpenGL display lists (layers)

Graphical output is done within a separate graphics window of MAKROS.

Element models can also be transferred to AutoCAD and displayed or modified within the active AutoCAD view.

The following information can be visualized:

currently defined macro model

currently defined finite element model

external element ID's, group ID's, mechanical type ID's, orientation of the elements etc

specified subdivision of the edges of the macro elements

external ID's of the nodes

boundary conditions and loads of the finite element model

distribution of scalar based results

distribution of vector based results

simulation of dynamically oscillations

The element model can be displayed as a wire frame model, a wire frame model that shows only sharp edges and as a surface model. The surface model can be rendered and shaded. Parts of the model can have different colors and can be displayed in different layers.

Graphics of MAKROS uses OpenGL functions of Windows NT and Windows 95 where parts of the graphics are internally stored within display lists. The use of such display lists makes it possible to vary aspects of the current view dynamically without a new calculation of the graphics. This way a high performance in the graphics display can be achieved.

Analogous to the notations within AutoCAD we'll use within the following sections the term „layer" also for these OpenGL display lists. You can assign different layer IDs to different parts of the structure and names to these layer IDs, so structural parts can easily be identified by name. In case a structure is read from AutoCAD all layer information and layer colors are maintained. The assignments of layers and layer names will also be saved to hard disk.

In case the structure is plotted within several layers, it's also possible to show or hide these layers within OpenGL.

For the definition of the 3D views within the OpenGL window two dialog boxes are available. With sliders rotation and zooming of the structure can be done quickly.

Note: For OpenGL graphics the color palette should be set to „True Color" (see Windows menu Display/Settings/Color Palette).

Filenames

Saving the data to hard disk (command "**Save**") uses several files with fixed extensions. These extensions are:

*.mes:	binary file of the macro model
*.mut:	binary file with the data to subdivide the macro elements
*.fes:	binary file of the finite element model
*.efp:	ASCII-file with node coordinates
*.efe:	ASCII-file with element descriptions
*.lqd:	binary file with loads, constraints, element properties and material data

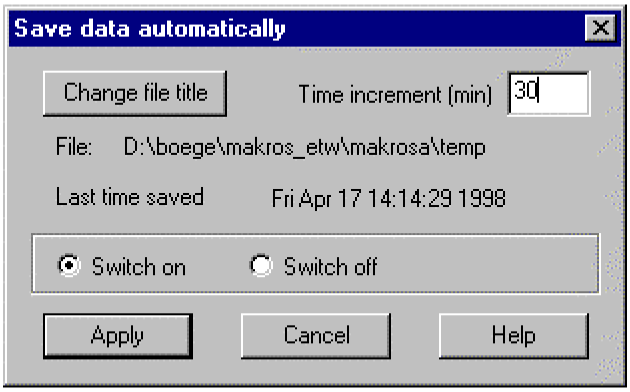
*.pos:	binary file with post processing data
*.ews:	saving of element selection sets
*.btx:	saving of text blocks
*.dem:	demo files

When a filename is required, a file selection window pops up where an existing file can be selected or a new filename can be given. The basis name of this given filename is only used, the filename extension is automatically appended according to the data being saved. Only files with appropriate extension are shown in the file selection window.

Automatically saving of all data

With the command „**Save automatically**“ a time increment and a basis name of the files for automated saving can be given. The basis name is appended by _1 respectively _2 and the filename extension according to the type of data being saved. Data is saved alternately in files with expansion _1 respectively _2, so always two copies of different times are available. Saved are the macro model, the finite element model, subdivision parameters and load data that is currently in memory. In case of a mistake, you should turn of the automatically saving immediately, so that a correct saving is not overwritten.

Parameters are to be given in the following dialog box:



Change file title: In a file selection box the basis name of the files is to be specified

Time increment: The time increment for automated savings is to be given in minutes. The time of the latest saving is displayed in the dialog box.

Switch on: Automatically saving is switched on.

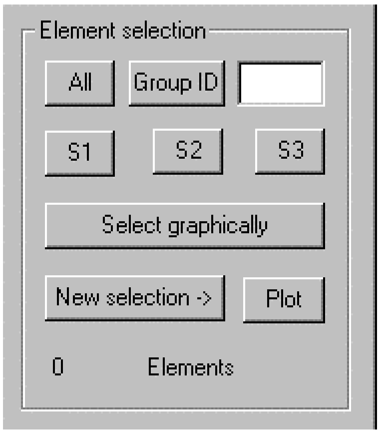
Switch off: Automatically saving is switched off.

Undo function

Most dialogs, that generate new nodes and elements, have an „Undo“ button. Recently generated elements are plotted in yellow and can immediately be deleted by clicking this button, this has to be done before any other operation is given

Element selection

Commands, that influence only some elements have the following options for the element selection in the corresponding dialog box of the command:



All: All elements are selected

Group ID: The elements, that have the group ID given in the input field are selected. If two group IDs are given, these are interpreted as min- and max ID of an array of group IDs. If more than 2 IDs are given, these are interpreted as individual IDs.

S1, S2, S3: The element selection stored in selection sets S1 – S3 are used.

Select graphically: Clicking this option, you can select elements graphically with cursor.

New selection: Clicking this button, the dialog box for the element selection is popped up where other options for element selections are given.

Plot: Selected elements are marked with a symbol, so you can check the selection graphically.

Graphical selection of nodes and elements within OpenGL

Several commands have options for selecting nodes and elements graphically within the OpenGL graphics window.

After selecting such an option all graphically selectable nodes respectively elements are marked by a colored symbol. This symbol has then to be selected by cursor.

The graphical selection offers following methods, the cursor symbol indicates which method is active.

*Select individual nodes (key **e**):* Multiple nodes can be selected by clicking the left mouse button. Selection will be ended by pressing right (or middle) mouse button or pressing the key **r** respectively **p** to alter the selection method. The cursor symbol is a square.

*Rectangular area (key **r**)*: Opposite corners of a rectangular area should be marked. All nodes within this area are selected. The cursor symbol is a corner.

*Polygon area (key **p**)*: All the vertices of a polygon area have to be marked with the left mouse button. Pressing right button or the keys **e** respectively **r** closes the polygon. The polygon is plotted. All nodes within this polygon are selected. The cursor symbol is a rubber band line.

During a single selection process several individual nodes or nodes belonging to distinct rectangular areas or polygon areas can be selected. The selection is ended by pressing the right mouse button.

The kind of selection can be altered during the selection process by pressing one of the following keys:

e: Change to selection of individual nodes

r: Change to selection of rectangular areas

p: Change to selection of polygon areas

Key input during the selection process will be recognized only when input focus is within the graphical window.

When selecting individual nodes the view can be varied before each selection. If there are more than one selectable point in the square area around the pointer in case of wire frame plot, this point is used that is nearest to the cursor position. In case of hidden surface plot, there may be some invisible points near the pointer, therefor in this case, the point is used that is nearest to the camera position. Note: If there is no visible point in the square, an invisible point may be selected. With hidden surface plot only selection of individual points is allowed.

All selected nodes are marked by a colored symbol. In case of selecting a node by mistake this selection can be made undone by immediately pressing key **d** when the cursor is within the graphics window and the graphic window has input focus.

In case of a graphical selection is made active from within the dialog box for node or element selection the height of the square area around the pointer can be specified within pixel units in a pop up window.

External node and element Ids, Node elements

Node and elements get automatically an ID continuously as the are defined. Using command **Sort** Ids can explicitly assigned to nodes and elements. For the assignment of loads, restrictions and properties, the external Ids are used, for that reason it is preferable to use only Ids continuously beginning by 1, so that Ids can directly be used as indices. A not continuos succession of Ids is created, if some nodes or elements are deleted. By saving the data to a disk file, it is checked whether nodes and element Ids are continuously beginning by 1. If this is not the case, a warning is shown in the protocol window. Changing the Ids using command **Sort** it is checked whether a load file exists and it is asked whether the used Ids in this file should be adjusted.

For the definition of elements first nodes must be defined, to these nodes automatically a node element is assigned because nodes can only be stored in connection with elements. These node elements must explicitly be deleted if they are no longer needed. Saving the file to disk it is checked whether node elements exist and a warning is shown.

Demo files, Tutorials

Command sequences of MAKROS can be recorded to a file and later be reused for demonstration purposes. These demo files use the extension *.dem.

Several demo files are given on the installation CD and can be used as tutorials. Running these demos all dialogs associated with the invoked commands may also be shown with currently used parameter values. To get more details look up command „**Demo file**“ in the last chapter.

Demo installation

MAKROS will be periodically checked against an available license code provided by a dongle. In case there is no such dongle available, all the functions of MAKROS can be used without any restrictions, however no saving of files or transferring of data to AutoCAD is possible.

Element types

MAKROS differentiates between macro and finite elements. Macro elements serve the purpose of defining a rough geometry of the solid to be calculated. The macro model is in most cases generated within AutoCAD. After the definition of how the macro elements should be divided, a finite element model will be generated out of the macro model. Finite elements will be used as input to a finite element calculation.

Table 2.1 shows all available geometric element types.

The geometrical form of the elements is defined by a geometrical type ID consisting of 2 (3) digits. The first digit (except for digit 7) defines the number of corner nodes for the element:

1 = node element

2 = line element

3 = triangular element

4 = quadrilateral element

6 = pentahedron

7 = tetrahedron

8 = hexahedron

10 = plane surface with up to 10 straight or curved edges

40 = plane surface element with up to 40 straight or circular edges.

The second digit distinguishes the number of nodes on edges between the corner nodes and in the inner of the element surface:

0 = straight edge with no nodes between the element corners (exception type 400)

2 = curved edge with 1 node between the element corners

3 = curved edge with 2 nodes between the element corners

5 = curved edge with a variable number of nodes between the element corners

6 = curved edge with 1 node between the element corners and 1 node in the interior

7 = 2 nodes between the element corners and 2*2 nodes in the interior of the element surface (triangle elements only have 1 node in the interior).

As macro elements only the types x0, x2, x5 can be used, where x = 2,3,4,6,8,10,40. Finite elements can be used with all types except for x5 and 400. Using the command **Type2Type** (Chapter “Commands to generate new nodes and elements”) a conversion between some different types can be done where additional nodes between the element corners and in the interior can be generated automatically.

For each element following data will be stored:

external element ID

geometrical type ID

mechanical type ID

group ID

layer ID

color index

external ID’s of element nodes in the order shown in **Table 2.1a – 2.1h**

With mechanical type ID’s elements of identical geometrical shape (and type) can be distinguished due to their different mechanical behavior (for example plates and shells). The real meaning of different mechanical types depends on the used FE calculation program, (see e.g. NASTRAN interface).

With group IDs several elements can be grouped together, e.g. for displaying parts of the structure with different colors or to assign materials and loads.

The element types usually are automatically generated from AutoCAD entities or by subdividing macro elements. Should individual elements be defined numerically (command “Element definition”), following rules are to be followed:

For element types 10 – 85 first corner nodes of the elements have to be given, then intermediate nodes on element edges and in the end nodes in the inner of the elements.

Hint: Intermediate nodes on straight element edges may be given as 0 with macro elements, with finite elements only if the used FE program does this allow (transition elements).

In case of element type x5 the number of intermediate nodes on edges is variable. To mark the last intermediate node for an edge, this node gets a minus sign (except for type 105). 0 for an edge indicates that there are no intermediate nodes for this edge.

In case of element type 105 the number of corner nodes and intermediate nodes on edges is variable. For this type all nodes are given in continuous order as a polyline and all intermediate nodes on curved edges are marked by a minus sign. First and last node of the polyline must be identical. Between corner nodes several intermediate nodes can be given. In case of a single intermediate node a circular arc and in case of multiple intermediate nodes a spline curve will be generated for the corresponding edge. Elements of type 105 can have an object height, which is given by an additional node. The connection between this node and the first element node will be used as the first perpendicular edge for this element (see Table 2.1g). All other perpendicular edges are assumed to be parallel to this edge.

Elements of type 105 can be meshed freely or using a regular pattern for 4 node elements (see chapter “Subdivision of macro elements into finite elements“). For regular pattern these nodes, which serve as element corners must be known. The external IDs of these nodes can explicitly be specified during the definition of these elements. They will be saved after the node specifying the element’s height. The first node of such an element will always be used as the first corner; only the following 3 corner nodes must explicitly be specified. In case no corner nodes are specified, MAKROS uses for a subdivision with a regular pattern those 3 vertices with the smallest angles in addition to the first node.

Elements of type 400 are plane elements with up to 40 corners, all nodes are given in continuous order as a polyline and intermediate nodes on circular edges are marked by a minus sign (same as with element type 105). First and last node of the polyline must be identical. Only straight edges with 0 intermediate node and circular edges with 1 intermediate node are allowed.

Elements of type 105 and 400 may contain holes in the inner of the element. To define holes, one node of the border of the hole is connected with a node on the border of the element, so that a closed polyline can be defined containing al borders of the element (see example below).

With Command „**Element definition**“ it is possible to specify individual elements by giving the IDs of the element nodes. For example the elements shown in Table 2.1a – 2.1h are defined as follows:

Type 45: 1, 2, 3, 4, 5, 6, -7, -8, 9, 10, -11, -12

Type 85: 1, 2, 3, 4, 5, 6, 7, 8, 9, -10, 11, -12, 13, -14, 15, -16, 17,-18,19, -20, 21, -22, 23, -24, 25, -26, 27, -28, 29, -30, 31, -32

Type 105: 1, 2, 3, -4, 5, 6, -7, 8, 9, 1,

With object height (node 11) and explicit definition of corner nodes (nodes 1, 2, 5, 9), for the subdivision of the element with a regular pattern, the shown type 105 has to be defined as follows:

1, 2, 3, -4, 5, 6, -7, 8, 9, 1, 11, 2, 5, 9

Type 400 (with holes and circular edges):

1, -2, 3, 4, -5, 6, 7, -8, 9, -10, 7, 6, -13, 14, 15, -16, 17, 18, -19, 20, -21, 18, 17, -24, 15, 14, -27, 28, 29, -30, 31, -32, 29, 28, -35, 4, 3, -38, 1

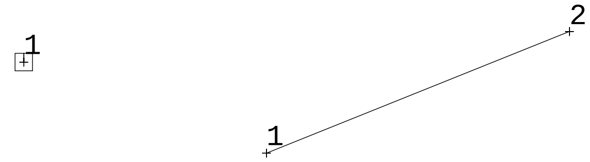
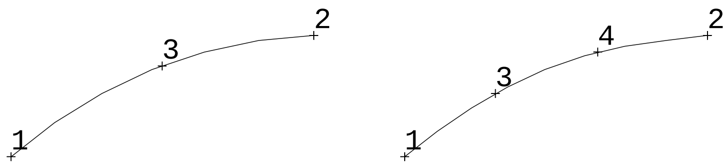
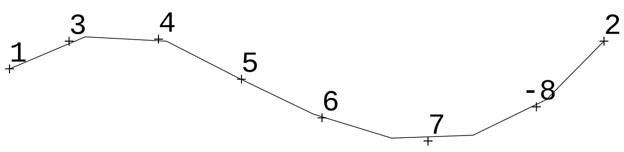
Type ID	Number of nodes	Geometrical shape
1 20	1 2-5	
22 23	3-5 4	
25	<=10	

Table 2.1a Types of macro and finite elements

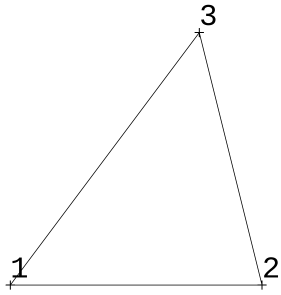
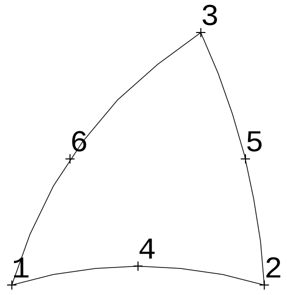
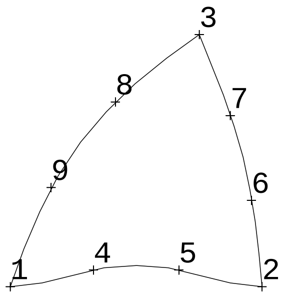
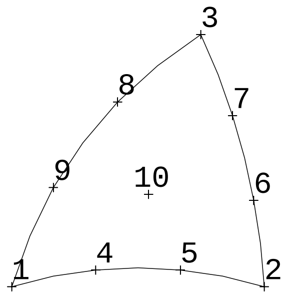
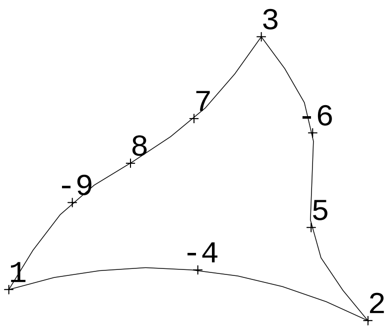
Type ID	Number of nodes	Geometrical shape	
30 32	3 6		
33 37	9 10		
35	<=30		

Table 2.1b

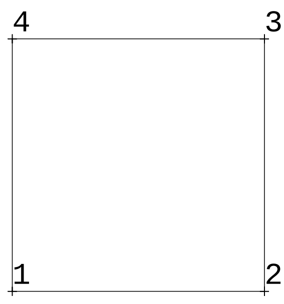
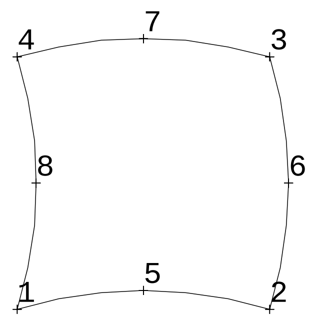
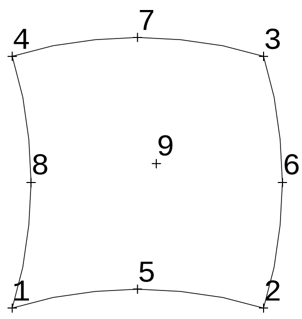
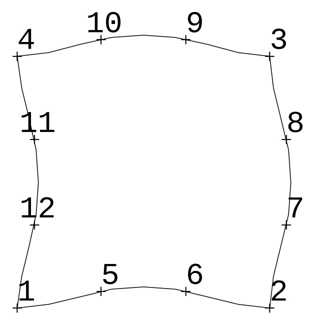
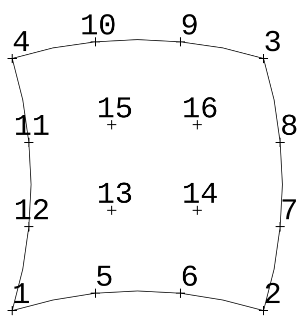
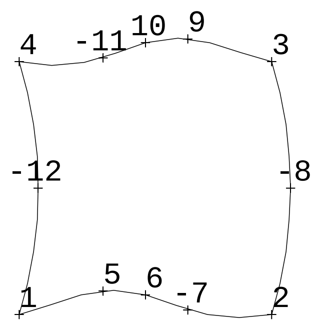
Type ID	Number of nodes	Geometrical shape	
40 42	4 8		
46 43	9 12		
47 45	16 <=40		

Table 2.1c

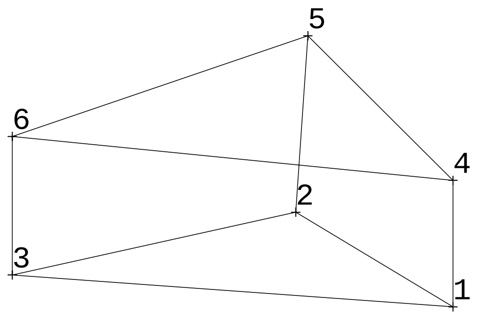
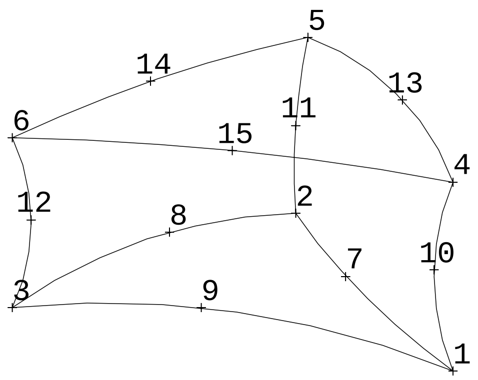
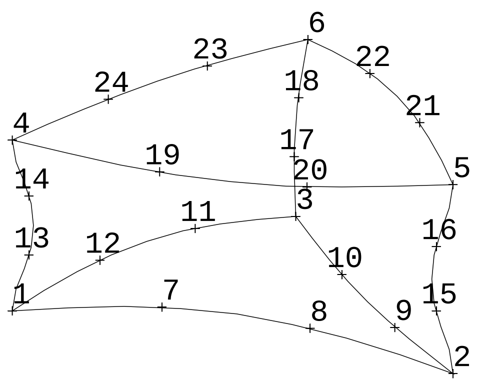
Type ID	Number of nodes	Geometrical shape
60	6	
62	15	
63 65	24 <=40	

Table 2.1d

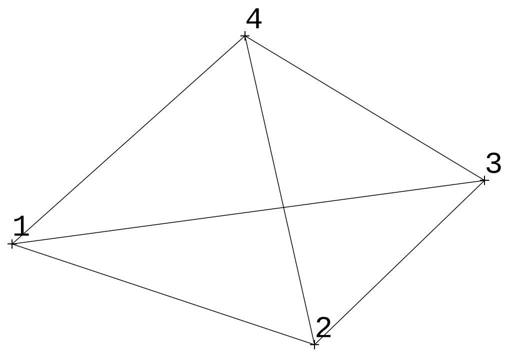
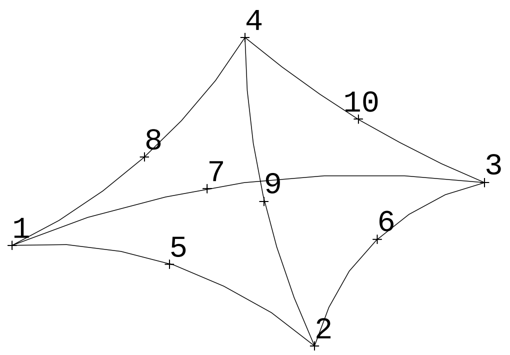
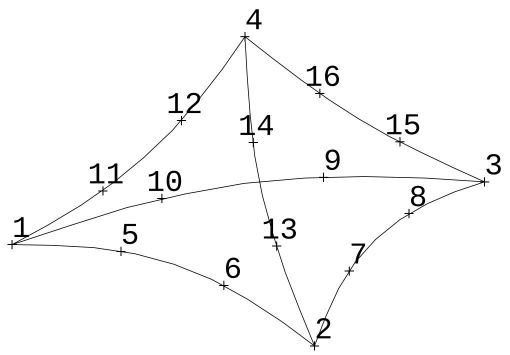
Type ID	Number of nodes	Geometrical shape
70	4	
72	10	
73 75	16 <=40	

Table 2.1e

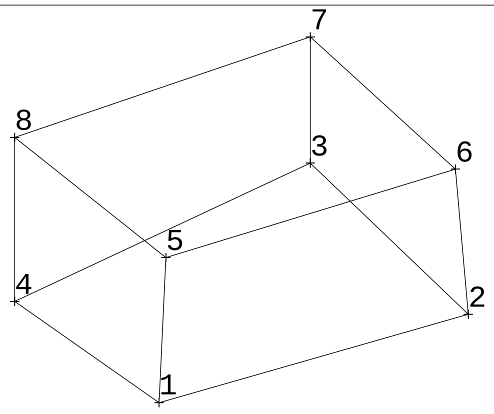
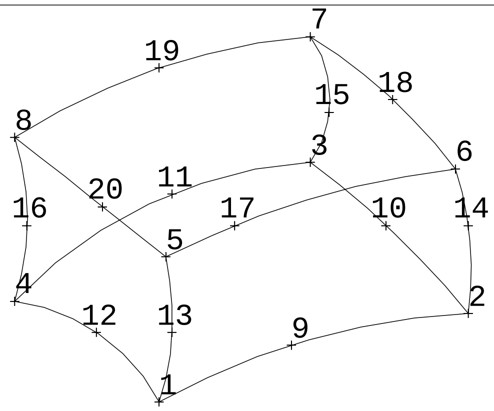
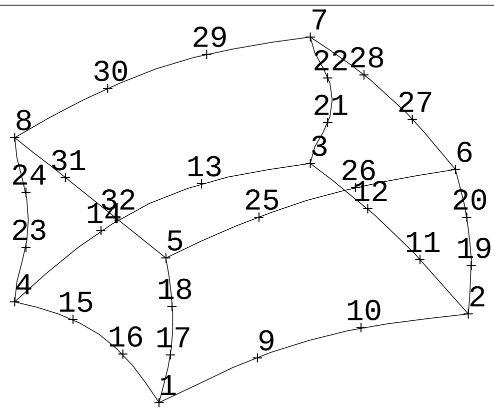
Type ID	Number of nodes	Geometrical shape
80	8	
82	20	
83 85	32 <=40	

Table 2.1f

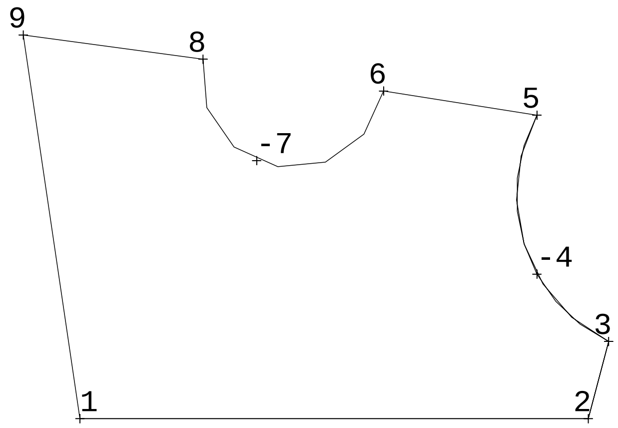
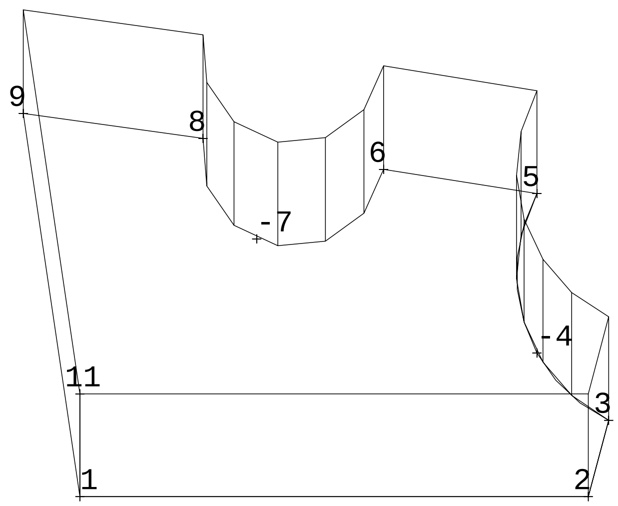
Type-ID	Number of nodes	geometrical shape
105	<=40	
105	<=40	

Table 2.1g

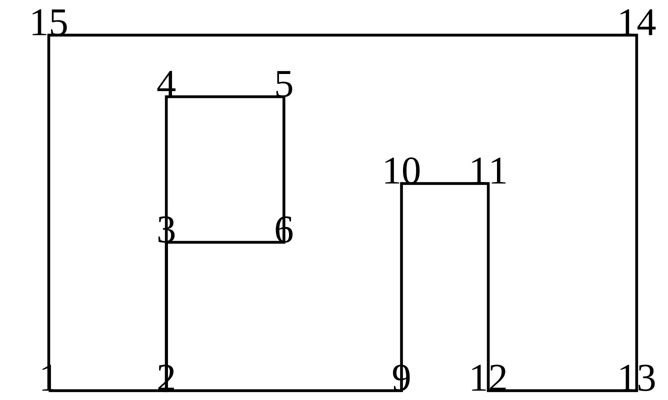
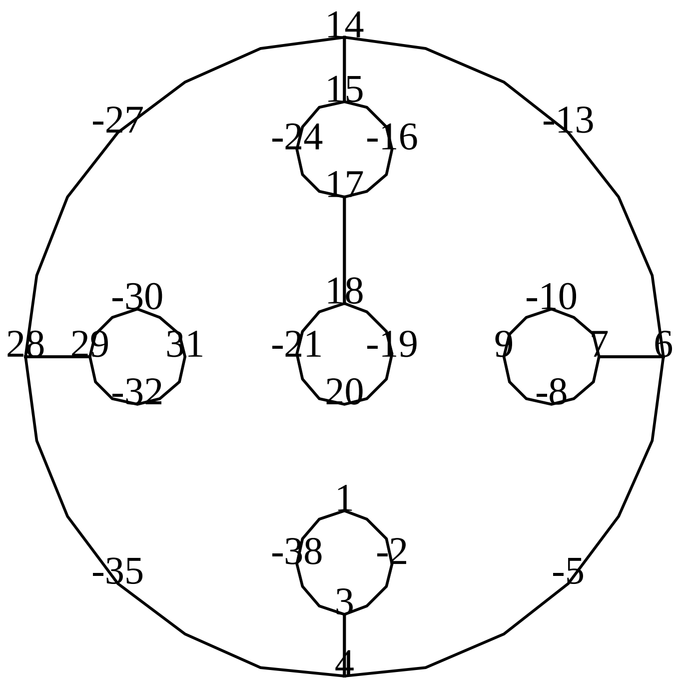
Type ID	Number nodes	Geometrical shape
400	<= 40	
400	<= 40	

Table 2.1h

Definition of macro elements within AutoCAD

The macro model should approximate the overall geometrical form of the solid to be calculated by use of large elements („macro elements“). It is important to specify the edges of the solid. The area between the edges is approximated by C^0-Coons surfaces. Macro elements are defined by corner nodes and intermediate nodes on edges. For each edge it is possible to specify 10 nodes, for each surface element no more than 20 nodes (40 for type 105 and 400), for each solid element no more than 40 nodes can be specified. Each edge consisting of more than 1 intermediate node between corners will be approximated by a spline curve. Edges with exactly 1 intermediate node will be approximated by a circular arc.

The basis of the element structure will be an AutoCAD generated construction model. For the definition of macro elements it is a good practice to construct additional cross sections of the construction model and to generate additional nodes placed on model edges and intersection edges.

Macro elements can easily be defined following these rules:

Line elements

Line elements are defined with planar or three-dimensional polylines. Polylines having only 2 vertices will be converted to element type 20, not closed polylines with more than 2 vertices are converted to type 25.

Surface elements defined by closed polylines

The easiest way to define surface elements is by defining planar or three-dimensional closed polylines comprising element corner nodes and intermediate nodes.

Two strategies are alternatively used by MAKROS to distinguish between element corner nodes and intermediate nodes on element edges:

1) Succeeding vertices with equal coordinates (denoted as “double vertices”) are used as element corner nodes. Vertices with a small distance (less than 1/50 of the longest distance between two vertices) are also regarded as double vertices.

2) The angles between succeeding edges are checked. Vertices with smallest angles are used as element corner nodes.

The interpretation of a closed polyline got from AutoCAD follows these rules:

First vertex of the polyline defines the first element node.

The polyline is considered to be closed, if the distance between the first and the last vertex is less than 1/50 of the largest distance between two vertices, or if the polyline is marked as closed by AutoCAD.

Polylines with 4 vertices will be converted to type 30 elements; polylines with 5 vertices will be converted to type 40 elements.

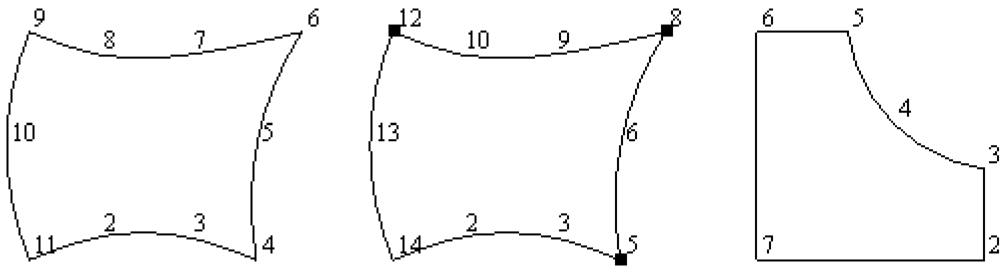
Polylines with more than 5 vertices are alternatively converted to type 35 or 45 elements (option T45), to type 105 elements (option T105) or to type 400 elements (option T400), the option must be marked in the corresponding dialog box. With option T400 marked, an element of type 400 is generated with only straight edges; if some edges should be circular, the intermediated nodes on these edges must later be marked with a – sign using command “Element definition”.

To find out the element corner nodes for elements with curved edges, first the presence of double vertices is checked. If double vertices are found, these are interpreted as corner nodes of the element. In case of the presence of a double vertex, all corner nodes of the element must be defined using double vertices. 2 double vertices within the polyline define type 35 elements, 3 double vertices define type 45 elements and more than 3 double vertices define type 105 elements.

If no double vertices can be found within the polyline, the angles between the 2 edges of each vertex are calculated and the vertices with the smallest angles are used as corner nodes. The angles have to be less than 150 degrees (another value can be given within the dialog). If 3 or more vertices with angles less than 150 degrees are found, element type 45 will be generated (where these 3 vertices with the smallest angles are used as corner nodes). With 2 vertices found type 35 elements will be generated. Polylines with more than 3 vertices with angles less than a given value can optionally be stored as type 105 or type 400 elements. This option must explicitly be selected in the dialog.

Between the element corner nodes it is possible to define 0 or more additional nodes on the edges where you can use up to 8 nodes for each edge and up to 20 (40 for type 105 and 400) nodes for the complete element (without double vertices).

Example: In the following figure the left and right elements are defined by a simple polyline where the smallest angles are used to fix the element corners. The second element is defined by a polyline where double vertices are used to fix the element corners. The vertices of the polylines are numbered continuously; a circle marks double vertices. The left polygon consists of 11 vertices, vertices 4, 6, 9 will be used as element corners because these vertices have the smallest angles. The second closed polyline consists of 14 vertices with vertices 4/5, 7/8 and 11/12 with identical coordinates, so these vertices are used as element corners. The right polyline consists of 7 vertices, where the angles in 4 internal vertices are less than 150°; a type 150 element is created because the option T105 is set in the dialog box.



Surface elements defined by closed polylines

In case of exactly one intermediate node on each edge type x5 element will be converted to type x2 element. For exactly 2 intermediate nodes, type x5 will be converted to type x3 element (for FE model only).

For element type 105 and type 400 all vertices must be defined within one plane. We suggest defining these elements as 2D polylines within a local coordinate system.

AutoCAD surfaces

Several surfaces generated by AutoCAD can be used and directly converted to macro elements. These are the types 3D-surface and 3D-polygon mesh.

3D surfaces will be converted to elements of type 40 or 30.

3D polygon meshes will optionally be converted to type 40, 42 or 45 elements. You can specify, whether all or only some mesh lines are to be used for the generation of the elements. The mesh can be single or double closed.

Solid elements defined by multiple closed polylines

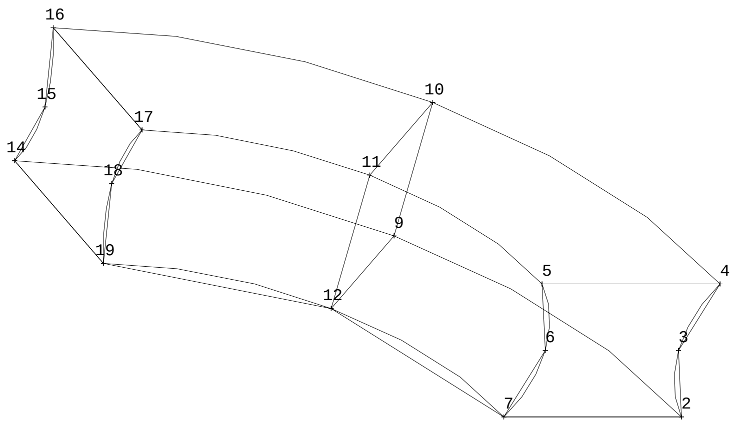
Solid elements are defined by a polyline consisting of at least 2 internal closed polylines. The interpretation of such a polyline generated by AutoCAD is based on the following rules:

Starting from the first vertex the next vertex within a distance of less than eps to the first vertex will be searched (vertex k1, vertex 7 in following figure). If this is the last vertex of the polyline, a surface element will be generated. The polygon area described by these vertices will be interpreted as a surface element of type 3x or 4x (x = 0 or 5). This surface element will become the bottom face of the solid element. The vertex following k1 will be used as the starting vertex for the next closed polygon area and the next vertex within a distance of less than eps to this vertex is searched (vertex k2, vertex 12 in the figure). This area will be considered as a cross section of the solid element (or the top surface of the element, if no further closed polygons follow). Zero or more cross sections between the bottom and top surface can be given.

If the solid element is only built up with straight edges, the fourth and the first vertex and the fifth and the eighth vertex for element type 60, the first and the fifth vertex and the sixth and the tenth vertex for element type 80 are identical. These possibilities are checked first. Type 65 and 85 can consist of up to 8 additional nodes on edges. The total number of nodes must be no more than 40 nodes.

Solid elements can also be defined within AutoCAD as closed 2D polylines with a given object height. This will result in a solid element with constant height where the polyline defines the bottom surface. In case of 4 or 5 polyline vertices, elements of type 6x or 8x are generated. In case of more than 5 vertices, element type 105 is generated with an additional node indicating object height.

Following Fig. defines a solid element by a polyline with 19 vertices (continuously numbered) where vertices 1/7, 8/12 and 13/19 have identical coordinates. With this polygon a solid element of type 82 will be created.



Solid element defined by tree closed polylines

Commands processing the entire structure

Overview

The menu group „**Structure**“ contains commands that affect the entire structure.

The following commands are provided:

Activate FE-/Macro Structure	Switch between processing of FE respectively Macro model
Load file	Load elements from hard disk (binary file)
Save to file	Save elements to hard disk (binary file)
Save all	Save all data in memory to hard disk
Save automatically	Save all data automatically
Read AutoCAD data	Read elements from AutoCAD respectively from a DXF-file
Delete elements	Delete elements in memory
Delete all	Delete all data in memory
Macro <-> FE	Transfer elements between structure type

With command **Activate FE-/Macro structure** you can switch between processing the macro model and processing the FE model. With **Load file** a saved structure is read from a binary file, with **Save to file** the structure currently in memory is saved in a binary file. With **Read AutoCAD data** elements are read from a DXF file or from an AutoCAD session. Command **Macro <-> FE** makes it possible to transfer elements from one kind of structure to the other.

Command descriptions

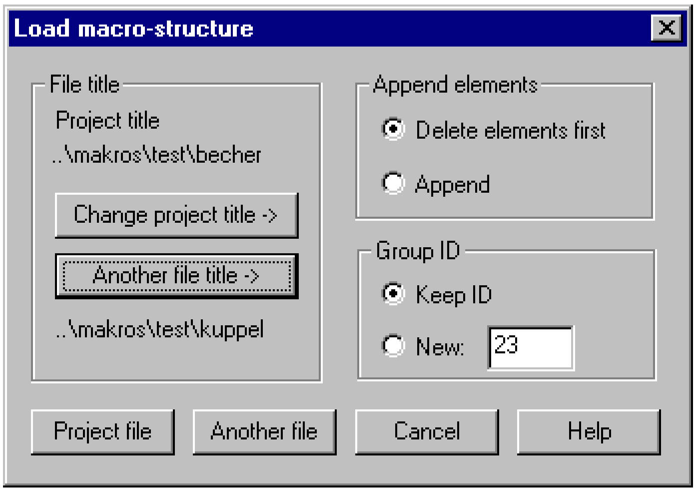
Activate FE-/Macro Structure: Switch between processing of FE respectively Macro model

Macro model and FE model can be loaded in memory simultaneously, but only one kind of structure is active for processing. The title bar of the graphics window shows, which structure is active. The command switches processing to the inactive structure. The label of the menu button shows, which kind of structure will be set active by clicking the button.

Hint: If memory is barely, only one kind of structure should be loaded at a time.

Load file: Load structure from hard disk

With this command a previously saved structure can be loaded. Following dialog shows the options available to specify the file title. Loading will be done by clicking button „Project file“ or button „Another file“:



File title

Project title: The name of the current project is shown; this file title is used if button "Project file" is pressed.

Change project title: A new project title can be selected from a file selection window.

Another file title: A file title can be selected from a file selection window; this title is used if button "Another file" is pressed.

Append Elements

Delete elements first: Before loading, existing elements in memory will be deleted

Append: New elements will be appended to existing elements

Group ID

Keep ID: Group ID of loaded elements is not changed.

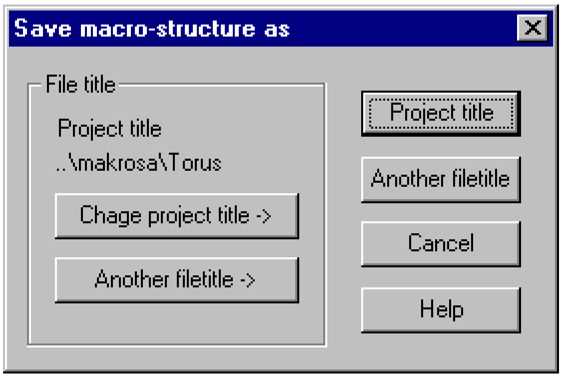
New: Loaded elements get the given group ID.

Save to file: Save elements to a file

This command saves all currently defined elements of the active structure in binary form in a file with extension .mes respectively .fes. When first saving, a dialog box is popped up where you can select a filename or create a new one. You can select any name. The file extension of the selection will be removed and replaced by the proper extensions. The title (without extension) will be shown as your project title within the dialog box. With „Change project title“ a new title for the project can be specified in a file selection window. With „Another file title“, a file title that is not the project title may be specified for saving.

Saving is done be clicking the button „Project title“ respectively „Another file title“.

Following dialog shows the available options:



Save all: Save all data in memory to hard disk

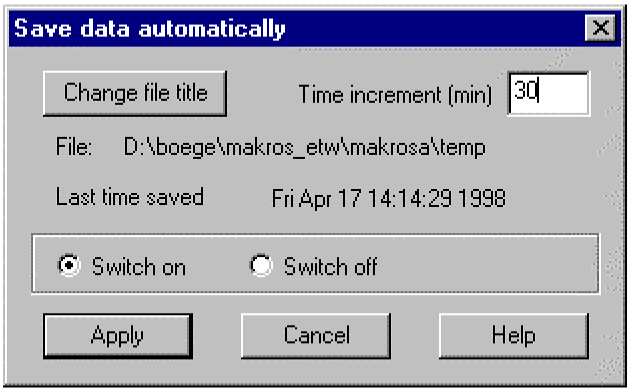
This Command saves all data in memory to hard disk files where the project title is used as a basis name. If no project title is specified, the title has to be given in a file selection window.

Attention: Existing files are overwritten without warning.

Save automatically: Save all data automatically to hard disk.

With this command a time increment and a basis name of the files for automated saving can be given. The basis name is appended by _1 respectively _2 and the filename extension according to the type of data being saved. Data is saved alternately in files with expansion _1 respectively _2, so always two copies of different times are available. Saved are the macro model, the finite element model, subdivision parameters and load data that is currently in memory. In case of a mistake, you should turn of the automatically saving immediately, so that a correct saving is not overwritten.

Following dialog shows the available options:



Change file title: In a file selection window the basis name of the files is to be specified.

Time increment: The time increment for automated savings is to be given in minutes. The time of the latest saving is displayed in the dialog box.

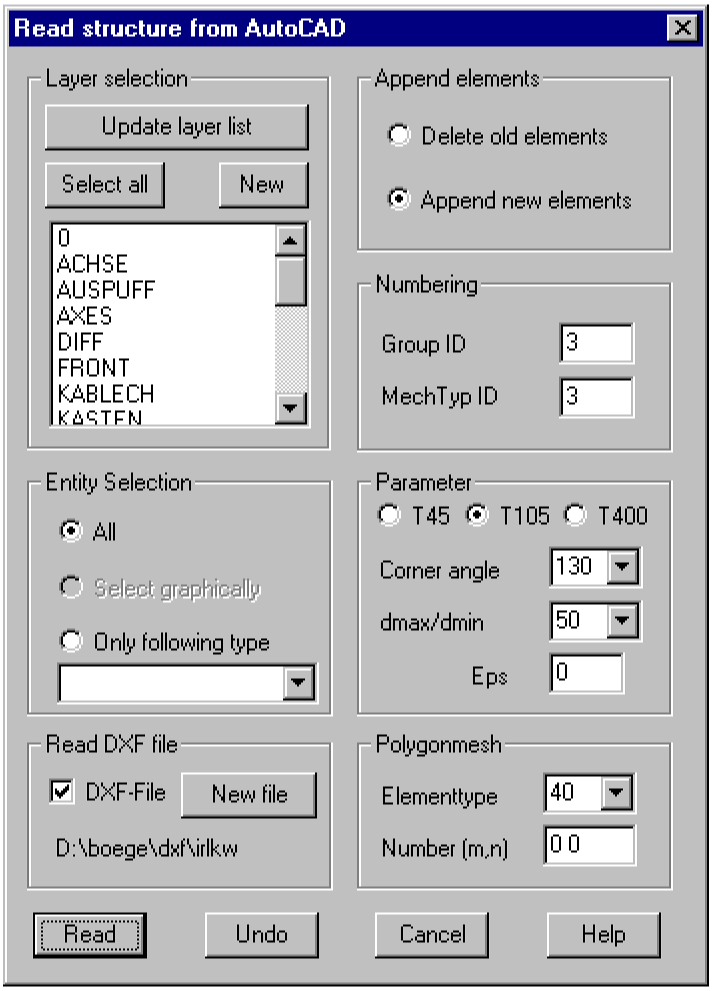
Switch on: Automatically saving is switched on.

Switch off: Automatically saving is switched off.

Read AutoCAD data: Read elements from AutoCAD or DXF-Files

With this command, elements can be read directly from a distinct or from multiple AutoCAD layers or from a DXF-File.

Chapter “Definition of macro elements within AutoCAD” describes which types of AutoCAD entities can be read. Only individual entities can be read, nested entities such as blocks and regions or polylines composed out of lines and arcs must be divided into individual entity’s first, using AutoCAD command “xplode”. If some entities cannot be interpreted as makro elements (e.g. only two corner nodes with a curved surface element), these entities are left out. Read elements will be appended to the prior read structure, except option "Delete old elements" is selected. After invocation of the command the following dialog box pops up. After checking and setting up the proper options the button "Read" will start reading.



Layer selection:

The list shows all available layers currently defined within AutoCAD or contained in the DXF file. Select the layers to read entities from. One or more layers can be selected. The button „Select all“ selects all layers and the button „New“ deselects all layers. With button „Update layer list“ the available layers are new checked, this could be necessary after new AutoCAD layers have been generated or a new DXF file is selected.

The AutoCAD layers are continuously numbered and this number is stored as a layer ID with the elements.

Entity selection

All: All AutoCAD entities of the given layers will be read and converted to macro elements. The interpretation of the entities is discussed within chapter “Definition of macro elements within AutoCAD”.

Select graphically: Only these entities will be read, which are contained within a following selection with AutoCAD commands. Only elements from one marked layer can be selected. Pressing Return ends selections. Start the AutoCAD selection after pressing button „Read“ and after you have confirmed that you will select graphically in a pop up dialog.

Only following type: If this option is selected, an entity type (POLYLINE, LINE, CIRCLE, ARC, SOLID, 3DFACE, TRACE, POINT) must be selected in the list. In this case only entities of the given type are read from the selected layers.

Read DXF-File

If the option „DXF-File“ is marked, the title of a DXF-File must be selected in a file selection window. The layers contained in the file are shown in the layer list. You can select individual layers and entity types that should be read from the file. Button „New file“ shows a file selection window where a new file can be selected.

Append elements

Delete old elements: All previously read elements will be deleted.

Append new elements: Newly read elements are appended to previously read elements.

Numbering

Group ID: You can specify a group ID for the new elements. Default is the highest group ID + 1.

Mechanical type ID: You can specify a mechanical type ID for the new elements. Default is the highest type ID + 1.

Parameter

T105: With this option selected, closed polylines with more than 4 corner nodes (vertices with an inner angle less than the given value) are stored as type 105 elements.

T400: With this option selected, closed polylines with more than 5 vertices are stored as type 400 elements with straight edges. If some edges should be circular, the intermediated nodes on these edges must later be marked with a – sign using command “Element definition”.

T45: With this option selected, polylines with more than 4 corner nodes (vertices with an inner angle less than the given value) are stored as type 45 elements, where only the 3 smallest corners and the first are considered as element corners.

Corner angle: Within this input field you may enter an angle value for the interpretation of polylines. All polyline vertices with inner angles lower than this value are considered as element corner nodes while remaining vertices in this case are considered as intermediate nodes on element edges. This is not applied by elements that are defined using successive polyline vertices with equal coordinates (double vertices).

Dmax/dmin: This input field may contain the ratio m. All vertices within a distance lower than dmax/m will be considered as double vertices. dmax stands for the maximum distance between 2 vertices.

Eps: After reading from AutoCAD all nodes with distance lower than this value (eps) will be merged into a single node. If no value > 0 is given, the value of eps will be set automatically in relation to the smallest edge length of all elements. Providing a value for eps would be useful to prevent merging of closely related nodes.

Polygon mesh

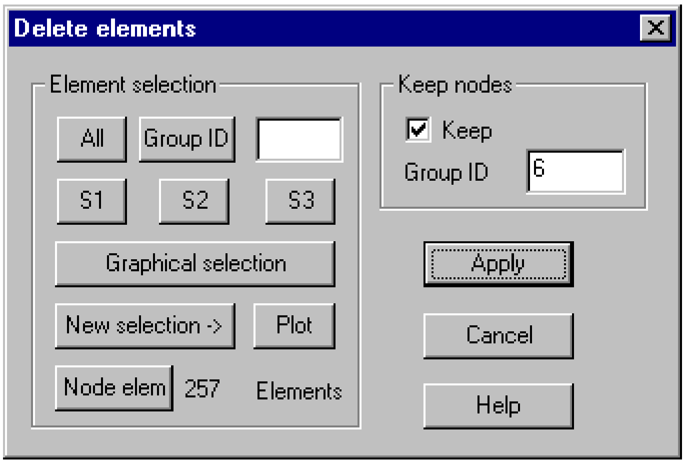
Element type: For polygon meshes it is possible to generate elements of type 40, 42 and 45. Default is saving each mesh as an individual 4-node element. With element type 42 2*2 neighbored meshes are saved as an element with one intermediate node on the edges. For element type 45 4*4 meshes are converted as an element with 3 intermediate nodes on the edges.

Number: Within this field it is possible to specify the number of elements to be generated for x and y direction. This means that not all edges of the polygon mesh will be used for element generation. If, for example, the polygon mesh consists of 128 segments in one direction and only every second edge should be used for generating type 45 elements, the number should be given as 16.

Delete elements: Delete elements in memory

Some or all elements of the active structure currently held in memory can be deleted. By default all nodes associated with these elements and not used by other elements are also deleted. However, by selecting option „Keep nodes“ these nodes are kept and may be reused later for element definitions. Nodes, which aren’t used by elements, will be associated with an element of type 1 (node element) and the given group ID, so these nodes are easy to identify. If those nodes are not needed any more, they can easily be selected by clicking button „Node elements“. Button „Apply“ will delete the selected elements.

Following dialog shows the available options:



Delete all: Delete all data in memory

This command is deleting all data currently in memory and frees allocated memory.

Macro <-> FE Append macro elements to finite element model or conversely

This command adds some or all macro elements to the finite element model or conversely. The element selection has to be done within a dialog. If the option “Append” is not checked, the inactive structure is first deleted.

Commands to generate new nodes and elements

Overview

Menu group „**Definitions**“ contains the following commands to generate new nodes and elements:

Define elements	Define individual elements graphically or numerically
Define nodes	Define new nodes, change node coordinates
3D-Extension/Translation	Transform surface elements to solid elements, Translate or copy elements
Mirroring	Mirror element structure on plane
Intersection	Calculate nodes on intersection of 2 regular surfaces
Beam element	Define additional nodes for beam elements
Net refinement	Local net refinement by dividing individual elements
Type2Type	Change element type
Coordinate system	Define local coordinate system
Subdivision	Create Finite Elements out of macro elements
DLL Function	Calling own functions from DLL

With **Define elements** and **Define nodes** individual elements respectively nodes are generated. **3DExtension/Translation** allows generating solid elements out of surface elements or to generate several layers of elements by translating one element layer or to move element groups to another position. **Mirroring** generates new elements by mirroring elements on a plain. **Intersection** allows generating nodes on the intersection curve of two regular surfaces. **Beam element** associates additional nodes to beam elements for eccentric connections and the main axis of inertia. **Net refinement** allows getting a local refinement of a plane FE net by subdividing individual elements. **Type2type** allows transforming one type of elements in to another type. Some commands process node coordinates in reference to local coordinate systems. Command **Coordinate system** defines these systems.

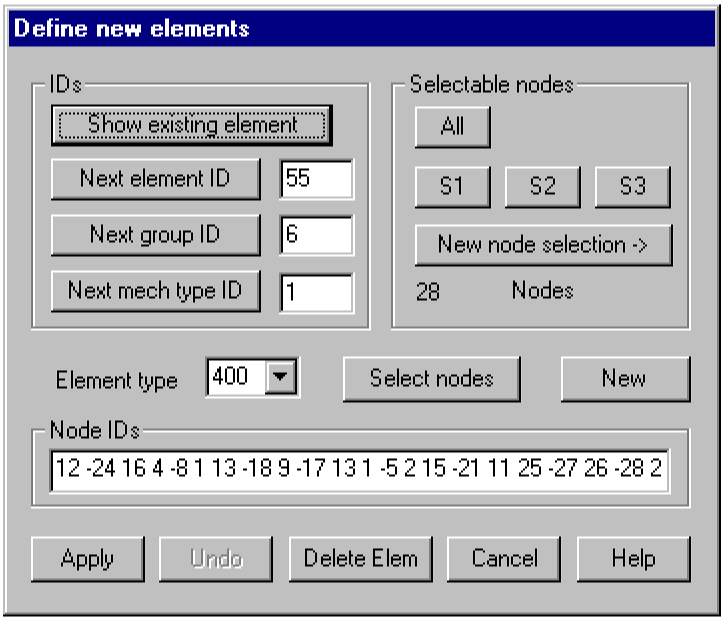
Command **Subdivision** contains functions to generate a FE net out of macro elements by subdividing the macro elements. These functions are described in chapter “Subdivision of macro elements into finite elements”.

Command description

Define element: Define individual elements graphically or numerically

This command lets you define individual macro or finite elements where associated nodes are already defined. When deleting elements an option must be set to keep the individual nodes. The dialog remains active until it’s explicitly closed by „Cancel“.

Following dialog shows the available options:



IDs

Show existing element: Clicking the button, an existing element can be selected graphically; whose parameters and node IDs are then shown within the dialog. This element can be modified and replaced by pressing „Apply“. For example it’s possible to add relevant nodes for a regular subdivision of a macro elements of type 105; or with type 400 elements to mark intermediate nodes on circular edges with a – sign, because this is not done automatically when reading type 400 elements from AutoCAD.

Next element ID: This input field specifies the element ID to be used for a new element. After each saving of an element, the currently largest element ID is incremented by 1. This ID will also be set if the button is pressed. In case the ID of an already existing element is given in the input field it will be asked for overwriting.

Next group ID: This input field gives the group ID for new elements. If the button is pressed the currently largest existing group ID incremented by one will be set.

Next mechanical Type ID: This input field gives the mechanical type ID for new elements. If the button is pressed the currently largest existing mechanical type ID incremented by one will be set.

Element type

Element type for new elements must be given. Only these types shown within the list box can be graphically defined.

Node IDs

Internal node IDs (indices of nodes) for the new element must be given according to the order shown in Table 2.1. For element type x5 the last node of a curved edge must be given a negative sign, while with type 105 corner nodes must be given a negative sign (see chapter “Element types“)

New

If this button is pressed, default values for the following available element ID will be set.

Select nodes

After clicking this button the nodes of the new elements are to be selected graphically. The IDs of selected nodes are shown within the dialog after finishing one element. The number of nodes to be selected depends on the element type. In case of a variable number (for example type 45, type 105 and type 400 elements) the selection is ended by pressing the right mouse button. Pressing the left mouse button does the node selection. In case the “-“ key is pressed first, the selected node gets a minus sign (last node for spline curves or intermediate nodes on edges for type 105 and type 400 elements). Selected nodes are marked by a colored symbol; by pressing „d“ key, latest selected node is made undone. In case there is a straight edge at elements of type x5, the node at this edge must be given as „0“, this is done with key ‘0’ when doing graphical selections. You can define new elements continuously until you press the right mouse button. Each new element is shown graphically and can immediately deleted by pressing button „Undo“.

Apply

This button saves a newly numerically defined element and immediately it will be colored displayed within the graphic window so its definition can be checked and it may be immediately deleted pressing button “Undo”. In case an already defined element should become overwritten it must explicitly be confirmed.

Undo

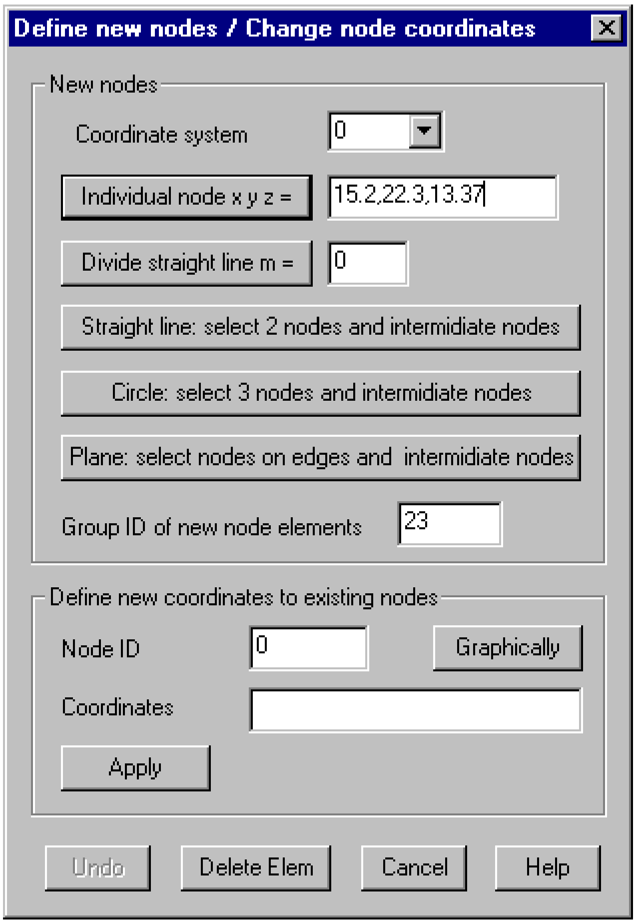
Pressing this button causes latest defined element to be deleted.

Delete elements

Clicking this button, the dialog for deleting elements pops up.

Define nodes: Define new nodes

With this command additional nodes used for later element definitions by command **Define Elements,** can be created. Node coordinates must be given within the following dialog. The dialog remains active until it’s explicitly closed by pressing „Cancel“.



Coordinate system

In case a local coordinate system is selected, all given coordinates are related to this system, applicable with options "Individual node" and “Divide straight line”.

Individual node

Coordinates in the selected coordinate system of an individual node have to be given in the input field.

Subdivide straight line

Pressing this button one or more straight lines are equally divided. The number of intermediate nodes on the line must be given in the input field. After pressing the button, 2 nodes as endpoints must be selected for each line. In case a local coordinate system is selected, the endpoints are transformed into the local coordinate system before the subdivision is done, so the new nodes are generated in this coordinate system. This way it is easy to get intermediate nodes on a cylindrical or a toroidal surface, but it is important, that the angle of the endpoints lie in the range 0° to 360°. Otherwise the new nodes might lie on the wrong half to the circle.

Straight line

Pressing this button, 2 nodes as end points of a line must first be graphically selected. A straight line is drawn colored. After that the positions of new nodes on this line have to be set by cursor (left mouse button). The right mouse button ends the creation of new nodes on the line, and a new line can be selected until the right mouse button is pressed twice.

Circle

Pressing this button, 3 nodes on a circle have to be selected first. The circle will be drawn colored. After that the positions of new nodes on the circle have to be set by cursor. The right mouse button ends the creation of new nodes.

Plane

Pressing this button, several existing nodes on the edges of a plane area must first be selected graphically. These edges are plotted as an orientation. The first, second and last node of the edges are used to define the plane, the angle between these node should be near 90°. The selection of nodes is interrupted with the right mouse button. After the edges are drown, the position of several new nodes on the plane can be specified with cursor until the right mouse button in pressed. For the graphic, parallel projection must be selected; otherwise the calculated position of the new nodes might differ from the cursor position. If the view is varied while new nodes are defined, the determining points of the plane must remain in the graphics window.

Group ID of new node elements

Nodes created by this command will be assigned a node element. Within this input field the group ID for these elements can be given.

Define new coordinates to existing nodes

Node ID: The ID of an existing node must be given. Pressing button “Graphically” a node can be selected graphically by cursor, the ID and coordinates of the selected nodes are shown in the dialog.

Coordinates: The new coordinates of the node must be given in the input field.

Apply: Pressing this button the new coordinates are stored in the data structure.

Undo

Pressing this button, last created new nodes are deleted.

Delete elements

Clicking this button, the dialog for deleting elements pops up.

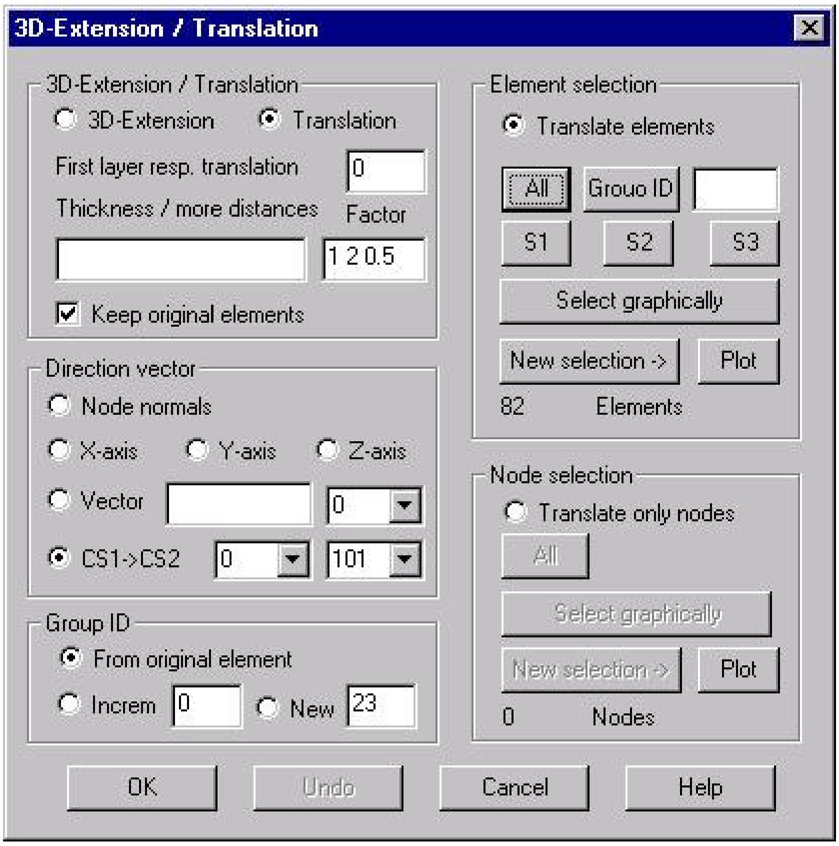
3D-Extension/Translation: Convert surface elements to solid elements, translate or copy elements

This command allows creating several layers of solid elements out of surface elements or to translate or copy elements.

Option „**3D-Extention**“ is to be used to generate solid elements out of surface elements. New nodes are generated in direction of the given vector. For that, the distance of the first layer of new nodes from the originating nodes has to be given with the thickness of one or more layers of solid elements.

If option „**Translation**“ is used, it is distinguished between the translation of elements and the translation of nodes. In the first case (option “Translate elements” checked) you can generate several layers of elements. New nodes are generated in the same way as with „**3D Extension**“, the distances of the first and following layers must be given. In the second case (option “Translate only nodes” checked) selected nodes are translated in direction of the given vector, without generating new nodes and elements. The amount of translation must be given in the input field “First layer resp. translation”.

Following dialog shows the available options:



3D-Extension/Translation

3D-Extension: Function for generation of solid elements will be selected.

Translation: Function for translation of elements or nodes will be selected.

First layer respectively translation: In the input field the distance of the first layer of new nodes from the originating nodes respectively the amount of translation in case of node translations must be given.

Thickness respectively more distances: For the generation of solid elements, the thickness of one or more element layers must be given. For the translation of elements the distances of the second and following layers must be given.

Factor: If elements are moved from one coordinate system to another (option CS1->CS2) scaling factors for the different coordinate directions may be given in this input field.

Keep original elements: With this option set the original elements are kept, otherwise they will be replaced by the newly generated elements.

Direction vector

Node normals: The direction of 3D extension respectively translation is determined by the mean value of all surface normals having this node as a vertex. Prior should be tested whether all elements are oriented the same way. This option is only implemented for elements of type 30 or 40.

X-axis: 3D extension respectively translation in direction of x-axis.

Y-axis: 3D extension respectively translation in direction of y-axis.

Z-axis: 3D extension respectively translation in direction of z-axis.

Vector: The components of a vector giving the direction for 3D extension respectively translation can be given in the input field. In case a local coordinate system is selected, the coordinates are first transferred to this coordinate system and the translation of nodes is done in this coordinate system (for example normal vector (1, 0, 0) within a torroidal coordinate system).

CS1 -> CS2: This option allows to move element groups or to duplicate elements to another position. The selected nodes are first transferred to the left selected coordinate system and multiplied with given scaling factors, and then these coordinate values are used as coordinate values in the right selected coordinate system and transferred back to the global coordinate system. The global coordinate system must eventually be given using ID 0.

Element selection

Option „Translate elements“ is to use, if one or more layers of new elements are to be generated by 3D-Extension or by translation of elements. Selected elements are used as originating elements.

Node selection

Option „Translate only nodes“ is to use if nodes are to be translated without generating new elements. Selected nodes are translated in direction of the given vector.

Group ID

In case the original elements are kept it may be useful to assign new group IDs to the newly created elements. Following options are available:

From original element: Group ID of the originating element is kept

Increment: Group ID of the originating element will be incremented by this value

New: Elements will be assigned this given group ID

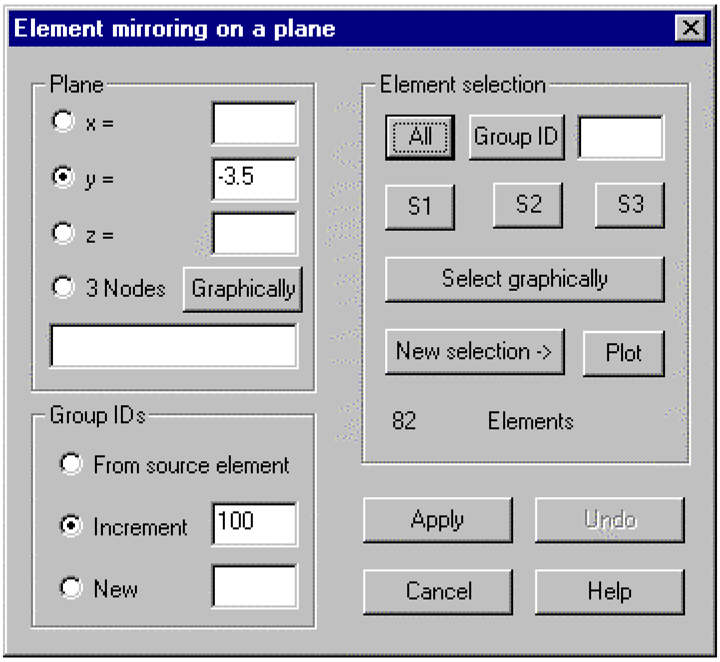
Undo

Clicking this button, the last newly generated elements are removed.

Mirroring: Mirror structure on a plane

This command mirrors parts of a structure on a symmetry plane.

Following dialog shows the available options:



Plane

<u>x =</u>: The mirror plane is parallel to the (y, z) coordinate plane of the global coordinate system. X-coordinate of the plane is to be given within input field.

<u>y =</u> : The plane is parallel to the (x, z) coordinate plane of the global coordinate system. Y-coordinate of the plane is to be given within input field.

<u>z =</u> : The plane is parallel to the (x, y) coordinate plane of the global coordinate system. Z-coordinate of the plane is to be given within input field.

<u>3 Nodes</u>: With this option checked, 3 nodes defining the symmetry plane have to be given or graphically selected after pressing button „Graphically“.

Group IDs

<u>Form source element</u>: New elements get same ID as originating elements

<u>Increment</u>: New elements get IDs based on originating elements incremented by this value

<u>New</u>: New elements get the given group ID

Element selection

Select elements to be mirrored. Elements of type 105 or 400 cannot be mirrored.

Undo

Clicking this button, the latest generated nodes and elements are deleted.

Intersection: Calculate nodes on the intersection curve of two regular surfaces

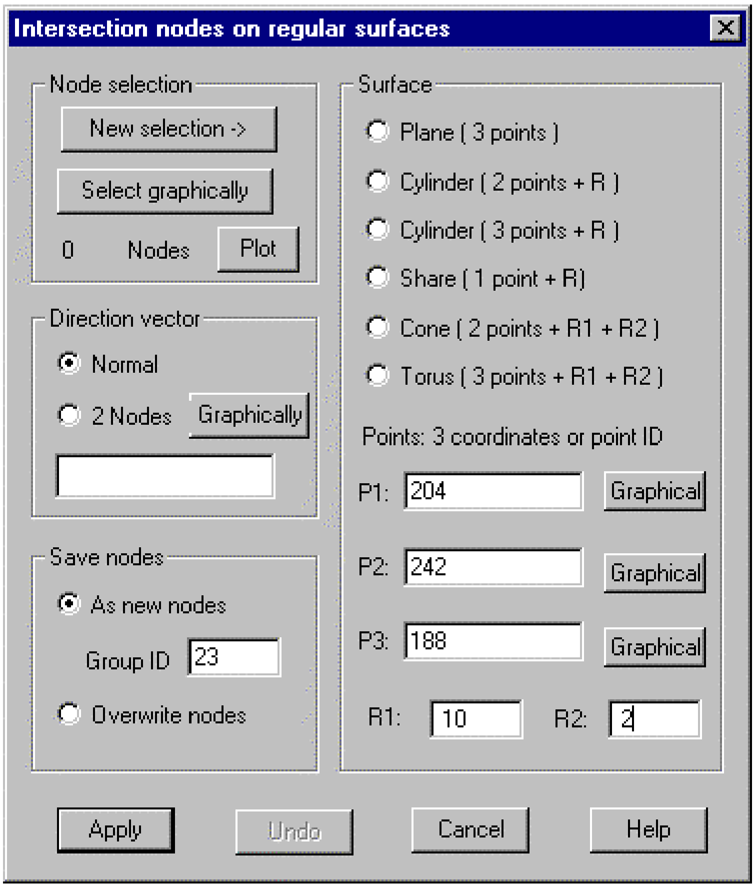
This command calculates nodes on an intersection curve of two regular surfaces. The following types of surfaces are supported: Plane, circular cylinder, sphere, cone and torus.

The command is used to produce individual nodes to define macro elements, or to move nodes created by the subdivision of macro elements, onto a regular surface to get a smoother structure. The definition is done by first creating additional nodes on one surface then move these nodes along a given direction onto the other surface. For example moving nodes on the surface of a circular cylinder along the direction of the cylinder axis onto a torus, will calculate nodes on the intersection curve of this circular cylinder with the torroidal surface. You can choose to create additional nodes or to move existing ones. Creating additional nodes will assign node elements (element type 1) to these nodes because only nodes of existing elements can be saved.

After specifying the parameters within the corresponding dialog the nodes to be moved onto the regular surface have to be selected. Button "Apply" causes the calculation to be done. Multiple selections can be specified. Pressing "Cancel" closes the dialog.

In case no intersection can be found, for example because the defining nodes of a plane surface are placed on a straight line or because there are no intersections within the given direction, an error message is produced. Otherwise the number of calculated intersections is shown.

Following dialog shows the available options:



Surface

Plane: 3 nodes defining the plane must be provided

Circular cylinder: 2 nodes defining the axis respectively 3 nodes defining a plane perpendicular to the cylinder axis and the radius of the cylinder must be provided.

Sphere: Center and radius must be provided

Cone: 2 nodes on the cone axis and the radii of the cone at these 2 nodes must be provided

Torus: 3 nodes within the plane of the torus and the 2 radii must be provided

Points P1-P3: Coordinates (x,y,z) of the related points or an existing node ID must be provided

By pressing „Graphical“ the corresponding node is to be selected within OpenGL window. The IDs of these selected nodes are entered within input fields.

R1,R2: If needed by the chosen surface, the radii must be provided

Direction vector

3 vector components of the desired direction of intersection or the external IDs of 2 existing nodes must be provided. In case of 2 IDs, the direction vector is build from the coordinate difference between these nodes. The direction can be arbitrarily, except with toroidal surfaces where the direction must be parallel or perpendicular to the toroidal plane. Clicking „Graphically“ causes the graphical selection of 2 nodes, defining the direction for intersection. In case option „Normal“ is marked, the normal vector of the surface is used as a direction vector, no additional input is necessary.

Save nodes

As new nodes: The calculated intersection nodes are saved as new nodes together with an associated node element. Within the input field "Group ID" these elements can be assigned a group ID.

Overwrite nodes: The coordinates calculated by the intersection replace the coordinates of the original nodes, i.e. the original nodes are moved onto the regular surface.

Node selection

Within the corresponding dialog nodes can be selected at multiple times. Nodes can also be selected graphically.

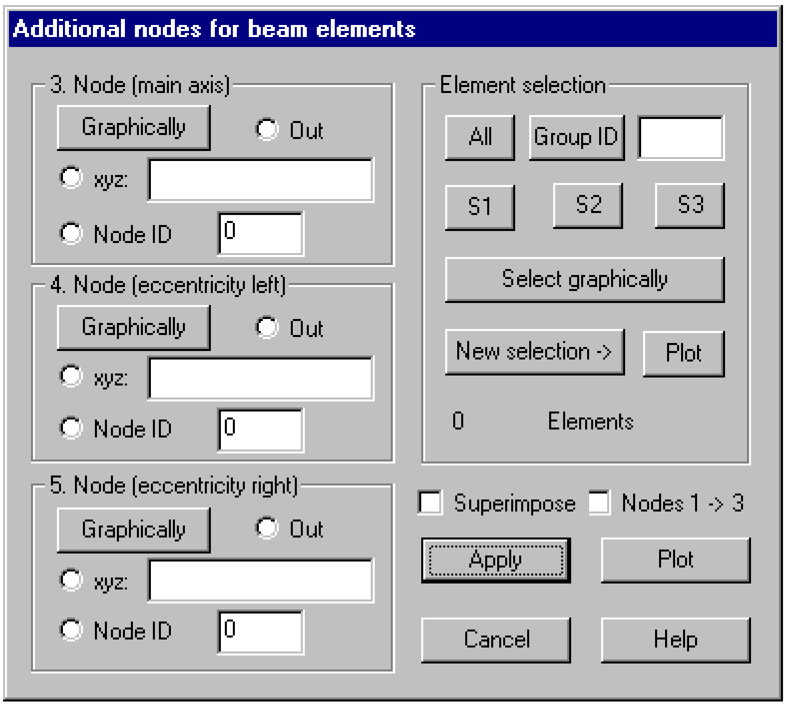
Undo

Pressing this button will delete the latest created nodes.

Beam element: Define additional nodes for beam elements

With this command elements of type 20 can be assigned up to 3 additional nodes. The first additional node (third element node) identifies the direction of the first main axis of the beam element. The forth node is used as an eccentrically connection of the left end node and the fifth node is used for an eccentrically connection of the right end node.

Following dialog shows the available options:



3. Node/ 4. Node / 5. Node

Graphically: After pressing this button, the node has to be selected by cursor. Node ID of the selected node is shown within the related input field.

Xyz: Within the input field coordinates for a new node to be generated can be given.

Node ID: The ID of an already defined node can be entered.

Out: If this option is selected, the corresponding node in the element definition is not changed.

Element selection

All these elements for which additional nodes should be defined have to be selected. Only elements of type 20 are considered.

Apply

This button assigns actual shown settings for selected elements

Plot

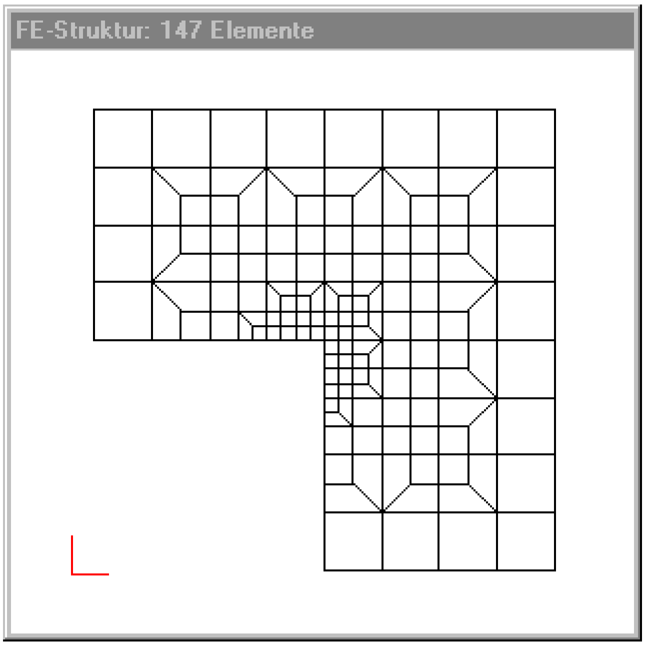
This button causes a plot of all additional nodes for actually selected elements of type 20. The main axis of beam elements is shown by a vector within center of element gravity in direction of node 3. If option “Nodes 1->3” is set, a colored line is plotted from nodes 1 to 3. The fourth element node will be connected to the left end and the fifth element node will be connected to the right element end by a colored line.

Net refinement: Local net refinement by dividing individual elements, respectively smoothing of a plane FE net.

This command is only applicable on plane FE nets consisting of only triangular and quadrilateral elements.

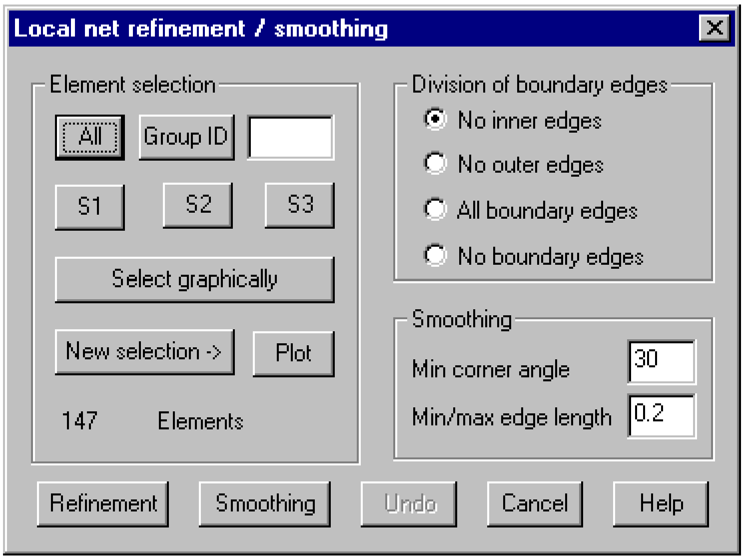
Refinement: Pressing this button, triangular elements are subdivided into 3 and quadrilateral elements are subdivided into 4 elements. One additional node is created in the inner of the elements and on the element edges of selected elements. In reference to the element edges that belong to the border of the selected area, it is distinguished between edges that belong to the border of the entire structure and element edges that do not belong to this border. It has to be specified, which of these two kinds of border edges should be subdivided.

Hint: The number of elements belonging to an element row next the inner border of the selected area should be even, so that only quadrilateral elements can be generated (see following figure).



Smoothing: Clicking this button, the selected area of the net is smoothed by using following algorithm: Successively, for each node in the inner of the selected area, the directly connected edges are determined and the coordinates of the node are corrected to the mean value of all nodes of these connected edges. The algorithm is used several times for each node. For smoothing a smallest corner angle and a ration for smallest to largest edge length of elements must be given. Correction of node coordinates is not done if the angle of element corners gets smaller than the given angle or greater than 180 - given angle, or if edge length gets to small.

Following dialog shows the available options:



Division of boundary edges

No inner edges: Element edges belonging to the border of the selected area but not to the border of the entire structure are not subdivided.

No outer edges: Element edges belonging to the border of the entire structure are not subdivided.

All boundary edges: All element edges belonging to the border of the selected area are subdivided.

No boundary edges: No element edges belonging to the border of the selected area are subdivided.

Smoothing

<u>*Min corner angle*</u>: An angle must be given in the input field, in case of smoothing the net, all angles in the corners of the elements are checked against this angle, and correction of the coordinates of some nodes is not done if an angle in the element corners gets less than the given value.

<u>*Min/max edge length*</u>: The ratio of smallest to largest edge length of elements must be given in the input field. In case of smoothing the net, all edge length of the elements are checked against this value, and correction of the coordinates of some nodes is not done if the ratio of edge length gets less than this value.

Element selection

Select the elements that belong to the area that is to be refined or that shall be smoothed.

Undo

Clicking this button will undo the refinement respectively smoothing

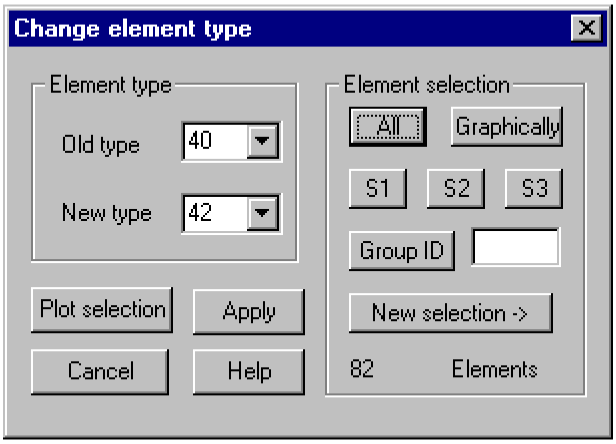
Type2Type: Change element type

Geometrical type for selected elements can be changed by automatically adding additional nodes on edges or on the interior of these elements by interpolation, or by deleting element nodes. By this way also quadrilateral elements with large distortions or large vertex angles can be automatically converted to two triangle elements (see command **Check elements**).

A type conversion may also be necessary after the creation of finite elements by subdivision of macro elements because the subdivision is only able to create one additional node on an edge.

Also type conversion may be necessary if elements with interior nodes are displayed within AutoCAD and later reread into MAKROS. In this case these interior nodes are lost.

Following dialog shows the available options:



Element type

<u>*Old type*</u>: The list box shows all currently existing element types in the structure. The type to be converted must be selected.

<u>*New type*</u>: The list box shows all possible new geometric types, to which the selected old element type can be converted. A new element type has to be selected.

Element selection

Only elements that have the given old geometric type and that are contained in the given element selection are converted. Button „Plot selection“ plots the element selection.

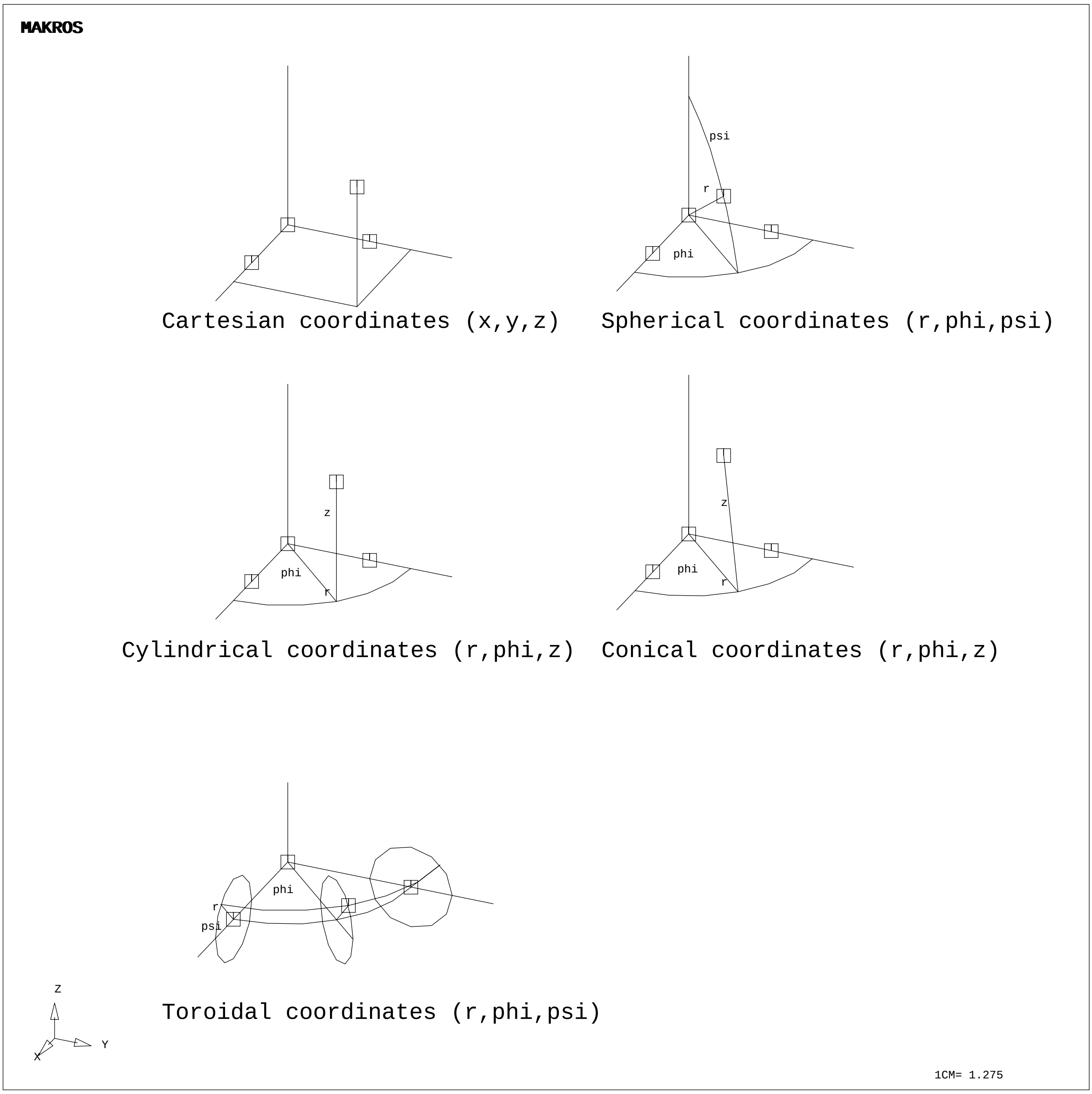
Coordinate System: Define local coordinate system

Different coordinate systems can be defined to be used later by commands like **Sort**, **Smooth, 3D-Extension/Translation, Boundary conditions**, **Nodal loads** . These coordinate systems are saved together with the nodes of the macro and the FE model. In case a new structure is loaded from hard disk they currently defined coordinate systems are maintained if they are not overwritten by loaded systems with identical IDs. No longer needed coordinate systems have to be deleted explicitly. The dialog shows the number of defined coordinate system.

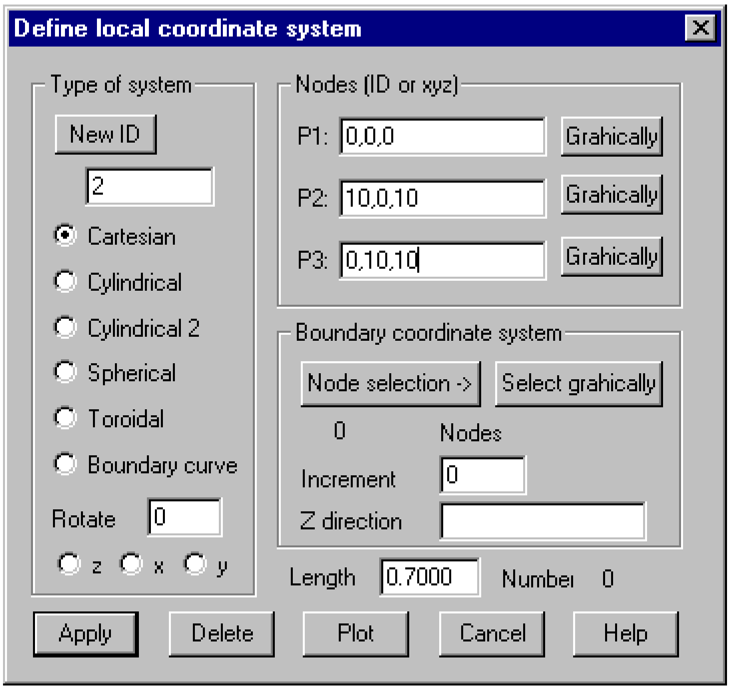
The orientation of coordinate systems is defined by three nodes, which can be specified either by a node ID or by global coordinates (x,y,z).

Node P1 defines the origin, P2 defines the positive direction of the local x-axis and P3 defines a node somewhere within xy-plane (except for „cylindrical2“ systems). The local z-axis is always set up to build a right-handed cartesian coordinate system. Toroidal coordinate system uses P2 as the origin, i.e. the distance P1-P2 defines the torus radius.

Following figure shows available coordinate systems and the definition of these systems.



Following dialog shows the available options:



Type of system

<u>*ID*</u>: Give the ID of the coordinate system to be created or range of IDs for coordinate systems to be deleted, plotted or rotated. Button “New ID” gets the next not used ID.

<u>*Cartesian*</u>: Cartesian system with coordinates x,y,z

<u>*Cylindrical*</u>: Cylindrical system with coordinates r,φ,z. Z-axis is perpendicular to plane P1-P2-P3

<u>*Cylindrical:2*</u> Cylindrical system with coordinates r,φ,z. Direction P1-P2 builds z-axis, P1-P3 build x-axis

<u>*Spherical*</u>: Spherical system with coordinates r,φ,ϕ. P1 = center node.

<u>*Toroidal*</u>: Toroidal system with coordinates r,φ,ϕ.

<u>*Boundary curve*</u>: This Option makes it possible to define multiple cartesian coordinate systems tangential to a given curve, for example for the definition of perpendicular or tangential loads or boundary conditions. The nodes of the curve must be selected within “Node selection“ dialog or graphically. For each node of the boundary curve the tangent vector is calculated. The x-axis of the coordinate system is oriented towards this vector and perpendicular to the given z-direction. The generated coordinate systems are numbered with the IDs of the related nodes plus the given increment. There are a maximum number of 500 coordinate systems that can be stored.

<u>*Rotate:*</u> This option makes it possible to alter the orientation of already defined local coordinate systems. The axis about which should be rotated must be selected and the angle of rotation must be given. The IDs of one or more defined coordinate systems can be given.

Nodes (ID or xyz):

Define points P1-P3 with node ID or global coordinates.

<u>*P1:*</u> The origin of the coordinate system.

<u>*P2:*</u> Node on x-axis.

<u>*P3:*</u> Node within xy-plane.

<u>*Graphically*</u>: The node is to be selected within the OpenGL window. The coordinates of these nodes will be updated within the corresponding input fields.

Boundary coordinate systems

When option “Boundary curve” is selected, the following additional parameters must be given:

Node selection: Node selection dialog pops up to define nodes for the boundary curve. The boundary curve is defined by all element nodes, which are contained within this selection.

Select graphically: Nodes of boundary curve will be selected graphically.

Increment: The increment for numbering the generated coordinate systems must be given (0 is allowed).

Z-direction: The direction of the z-axis of the local coordinate system must be given. This direction must not be within the plane of the boundary curve.

Delete

One or more coordinate systems are removed. The ID of the coordinate system or a range of IDs for several coordinate systems to be deleted must be given.

Plot

This button plots the coordinate systems who's IDs are given. Also the length of the axis must be given. ID 0 plots all coordinate systems. Newly defined systems are automatically plotted.

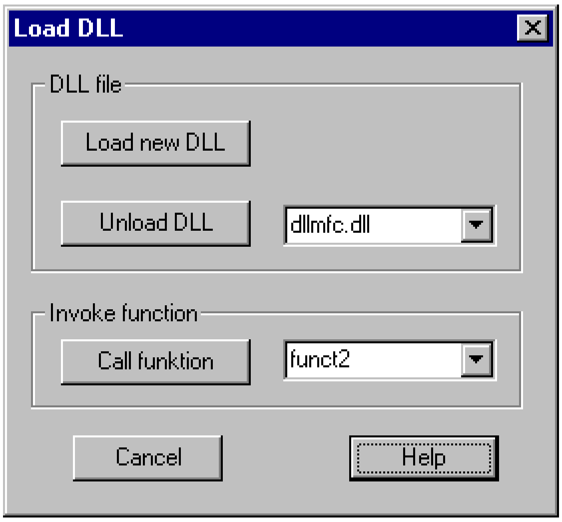
Subdivision: Create FE model out of macro model

This command contains functions to generate a FE net out of macro elements by subdividing the macro elements. These functions are described in chapter "Subdivision of macro elements into finite elements".

DLL function: Calling own functions from DLL

This command allows calling own functions for the generation of new elements. The functions must be provided in one or more DLL files.

Following dialog shows the available options:



DLL file

<u>*Load new DLL*</u>: Clicking this button, a file selection box pops up, where the file of the DLL to be loaded must be selected.

<u>*Delete DLL*</u>: All loaded DLLs are show in the list box. After selecting a DLL it can be deleted by clicking the button.

DLL function

<u>*Call function*</u>: A function name must be given in the input field. Clicking the button, this function will be called. If the function is called the first time, the name of the DLL where the file is contained in must also be selected in the list of DLL names.

The list shows all the functions that have previously called; these functions can be called several times.

When the command is invoked the first time, it is checked, whether there exists a file „maka_interface.ini" in the "bin" folder of MAKROS that contains some lines of following type:

#dll kf function path

„kf" is the type ID of the function, „function" is the function name of an exported function in the DLL and path is the path of the DLL, the function is contained in.

For example:
#dll 0 funct1 G:\boege\makrosa\testdll\dll1\debug\dll1.dll
#dll 0 funct2 G:\boege\makrosa\testdll\dllmfc\debug\dllmfc.dll
#dll 1 Interface G:\boege\makrosa\testdll\interf\debug\interface.dll

If lines of this type are found, the DLLs are automatically loaded and the entry points of the functions are searched and stored. The DLL and function names are shown in the list boxes of the dialog window.

Prototype of functions

All functions must have the following prototype:

```
typedef struct
{
	int ne;		// number of elements
	int  nn1;		// array length of one element
	int npv;		// number of nodes
	int **nel;		// elements: nel[nev+1][nn1]
	float (*pkt)[3];	// node coordinates
} festruct;

extern "C"
void function (CWnd *pWnd, festruct *in, festruct *out, int kfree, int kzf, void *p);
```

pWnd is a pointer to the protocol window of MAKROS. If the function uses dialogs, pWnd should be used as the parent window.

in is a pointer to a structure that contains the elements and the nodes of the active element structure. For each element following data is given: external element ID, type ID of the element, mechanical type ID, group ID and indices of the element nodes. Elements and nodes are stored beginning with index 1.

out is a pointer to a structure, that can be used to give new elements and nodes back to MAKROS. New elements are added to existing elements.

kfree is a parameter. The function is invoked twice, the first time kfree has the value 0, in this case new elements can be returned to MAKROS. After these elements have been added to the structure, the function is called again with kfree = 1, in this case only the allocated memory for the new elements should be freed.

kzf is the ID of the function type and must be 0 with this type.

p is a pointer to a structure and is not used with this type of functions.

Subdivision of macro elements into finite elements

Overview

The commands to subdivide a macro structure into finite elements are grouped together under a common dialog with property sheets. The dialog is made active by the command „**Subdivision**“ and remains active until it is closed by pressing „Cancel“. Pressing „Plot“ causes the current subdivision data to be plotted.

The following commands are available:

Division	Equidistant subdivision of edges
Pattern	Specify subdivision data for single elements
Edges	Specify subdivision for single element edges
Continuation	Continue with existing edge subdivisions
Reference point	Define reference points for subdivision of macro elements
Check	Check consistency for existing subdivisions
Delete	Delete currently defined subdivision data
File	Save/load subdivision data to/from file

Subdivision

Execute subdivision

The subdivision of macro elements into finite elements is done by applying various patterns. It is distinguished between regular subdivisions where opposite edges are subdivided identical and irregular subdivisions. The patterns are distinguished by a unique pattern ID.

The regular subdivision is selected by ID 20, 40, 40 for macro elements of type 2x, 4x, 8x and by ID 134, 30, 30 for types 3x and 6x (x = 0,2,5). The regular subdivision is set by default for all elements after initialization.

The irregular subdivision allows step-by-step refinements of the element mesh. There are several transition patterns available. Transition patterns are selected by IDs greater 100. The most flexible pattern is the pattern with ID 142 respectively 150, where all edges can have any number of subdivisions.

Solid elements only have transition patterns for the bottom surface. The third direction (z-direction) only can regularly be subdivided. The number of subdivisions for this direction is specified with the command **Pattern**.

For each pattern you'll have to define the number of subdivisions (number of intermediate nodes) for all edges of the bottom surface of the macro element in counter clock wise order. With solid elements also the number of intermediate nodes for the third direction must be given.

Marking the resulting intermediate nodes with a graphical symbol can graphically show subdivision data for edges. A shrink plot of elements may be especially useful to check the subdivision of adjacent edges. The main directions of macro elements will also be plotted to show the orientation of elements. This information will be needed for the command **Pattern** where the subdivision for individual directions of the elements must be specified.

For the specification of a subdivision the following commands are available:

Pattern specifies the subdivision data (pattern ID + number of intermediate nodes for element edges) for individual elements

Edges specifies subdivision data for individual edges of the structure where it's possible to have an automatic regular continuation for the complete structure or selected elements by assigning the same subdivision to opposite edges.

Continuation applies existing subdivision data for individual edges to opposite edges, similar to command **Edges**

Division assigns subdivision patterns in the way that the resulting edges of the generated finite elements are all smaller than the given value. It's also possible to specify that opposite edges get the same number of subdivisions (regular pattern).

Reference Point allows defining fixed points on edges that should be considered when the element is subdivided.

Check does consistency checks on the existing subdivision data. Edges are checked whether adjacent elements are equally subdivided. Optionally a given inconsistency can be automatically corrected.

Note: Consistency checks for adjacent elements require the use of identical nodes for common edges. Plotting the boundary edges of a structure can check this. By using the command **Compress nodes** (see chapter "General commands") it's possible to merge those narrow nodes into a single node.

The easiest way to specify subdivisions is by using the command **Division**. It's possible to specify different granularity's for different parts of the structure by using this command several times with different values for different element sets. Inconsistency between the parts must then be corrected with command **Check**.

To get a mostly regular subdivision to macro elements, following steps should be used:

Specify transition patterns for individual elements with the command **Pattern**.

Continue the subdivisions of the edges for adjacent elements

Specify a regular subdivision of all the remaining elements with **Division** and option “Continue”.

Check and correct consistency of subdivision data with **Check**

Note: The subdivision of macro elements only creates finite elements with straight edges or one intermediate node on edges. To get elements with two nodes between vertices or additional nodes inside the element, the command **Type2Type** has to be applied to change the element type and to generate additional nodes.

Note: The number of intermediate nodes on edges is limited to 60 nodes per edge. The structure may be subdivided in several steps by transferring a new created FE structure to a macro structure (command **Macro <-> FE**) and subdivide the new macro structure.

Note: If transition patterns are used some triangular elements may be created. To avoid this, first create Finite Elements with double edge length and subdivide these elements again by using a regular pattern with one intermediate node on edges.

Performing the subdivision of macro elements (Generating finite elements)

The subdivision is done by the command **Subdivision**

The calculation of additional nodes during the subdivision is done, on the basis of the assigned pattern, by a C^0-Coons interpolation of surfaces within the global coordinate system. In some cases, for example nodes on a spherical or cylindrical surface, it may be necessary to smooth the resulting nodes with the command **Smooth**. This will for example be the case when describing a cylindrical section by an element with spline curves.

The numbering of the generated nodes of the finite element structure is done by the following scheme: The 3 dimensional space where the entire finite element structure is contained within is separated into small cubes. The generated nodes will have a continuous numbering based on the cube they fall into where x-direction comes before y-direction followed by z-direction. With the command **Sort** a different numbering scheme may be applied.

Each individual macro element is subdivided into finite elements based on the previously defined subdivision data. Afterwards nodes on common element edges or surfaces or nodes with identical coordinates (within a given tolerance) are combined into only one node (see parameter eps for the command **Subdivision**).

Attention: Merging of nodes may be critical, if you don‘t give a value for eps, the chosen value is determined considering the length of all elements and may be to small. In any case check whether there are nodes within a small distance that should be merged into a single node by plotting sharp edges or by using the command **Compress nodes**.

The consistency of the generated Finite Element mesh can only be guaranteed if adjacent macro elements have identical edges with identical vertices and if subdivision data for these macro elements is also identical (edge related). This consistency will be checked automatically before doing the subdivision. Found inconsistencies will be shown on the screen. If any inconsistencies are found, it will be asked for further processing. The same mechanism can be used with the command **Check** where all edges with inconsistent subdivision data are graphically shown and these inconsistencies may optionally automatically be corrected.

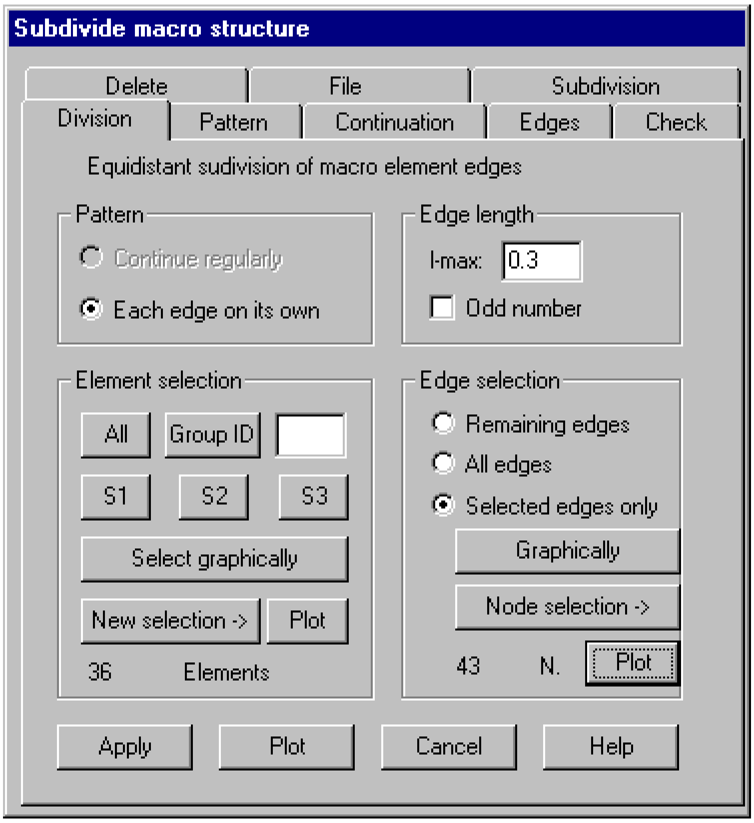
Note: For beam elements (type 2x) with an additional third node for defining the main section, this third node will also be present for all the generated finite elements.

Note: The subdivision of a macro element for a specific pattern is first done for a unit element. Afterwards the mesh is mapped to the geometry of the macro element. In case of large distortions the resulting mesh may become unpredictable so it may be useful to modify the macro structure to less distorted elements.

Command description

Division: Equidistant subdivision of macro element edges

This command defines a subdivision of the complete macro model, of individual elements or individual edges. After specifying parameters within the following dialog, button „Apply“ calculates the number of intermediate nodes for the edges and saves subdivision data.



Pattern

Continue regularly: Selected elements are regularly subdivided, that is opposite edges of an element will have identical numbers of segments (pattern 40 for 4-node elements). Starting from an edge with still undefined subdivision data, all edges are searched that will be affected by a continuation of this subdivision. From this set of edges the edge with the largest length will be evaluated and for this edge the number of segments is calculated and applied to all edges within this set. Regular continuation is only possible for 3 and 4 node elements. This option is not supported for elements of type 105 and 400.

Each edge on its own: The number of segments is calculated for each selected edge individually based on the edge length and the given element length “l-max”. The pattern 150 is applied. To quadrilateral elements with corner angles nearly 90 degree or triangular elements with corner angles with nearly 60 degree, pattern 142 is applied.

Edge length

L-max: The max. length for the edges of the finite elements must be provided. The number of segments for the edges of macro elements will be calculated based on this value. If there are reference points defined for an edge, the subdivision is done for the parts of the edge.

Odd number: This ensures that only an even number of segments will be generated by possibly increasing the number of segments by 1 (this isn’t applied in case of regular continuation). This will result mostly to get only 4-node finite elements und is especially important for type 105 and 400 elements, where subdivision is eventually first done with the double edge length and then all elements are subdivided into 4 respectively 3 quadrilateral elements.

Element selection

Select elements that shall be considered.

Edge selection

<u>Remaining edges</u>: All edges of selected elements with currently no specified subdivision data are selected.

<u>All edges</u>: All edges of selected elements are used. Old subdivision data will be overwritten.

<u>Selected edges only:</u> With this option a node selection must be given graphically or with the dialog for node selection. All edges which have both vertices contained within this set of nodes will be selected and subdivision data will be assigned only to these edges. Element selection will be ignored with this option. Not applicable with type 105 and type 400 elements.

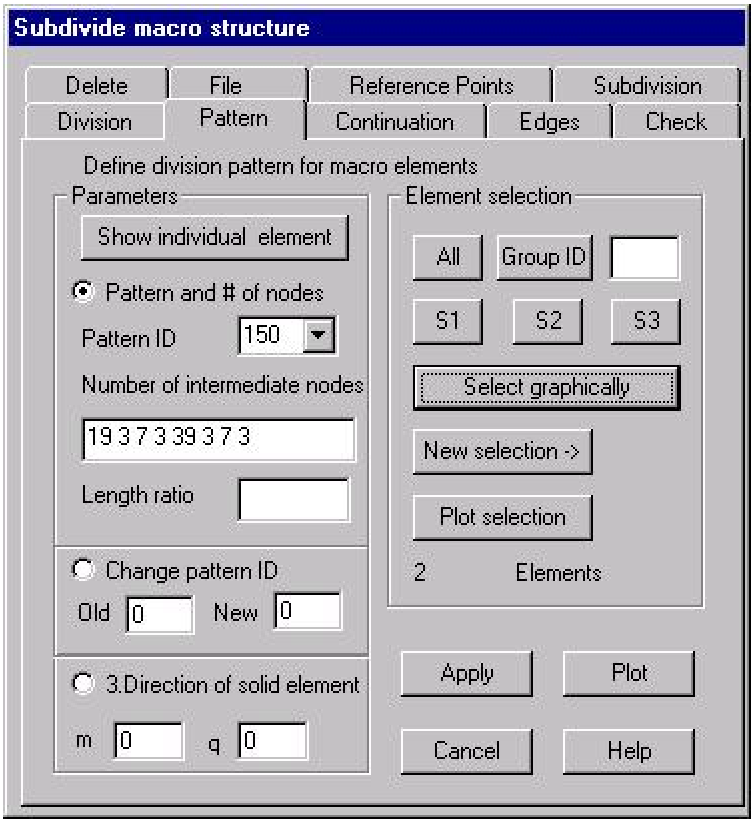
<u>Hint</u>: With solid elements, only subdivision data for the surface of the solid is defined. Subdivision for the third direction must be specified afterwards using command „**Pattern**"

Pattern: Specify subdivision data for individual macro elements

This command is used to assign special transition patterns to individual elements or to specify subdivision data for the direction perpendicular to the bottom surface of solid elements.

Button „Apply" assigns current settings to selected elements.

Following dialog shows the available options:



Show individual element

Clicking this button, one individual element can be selected graphically. The assigned subdivision parameters of the selected element are immediately shown in the dialog and may be changed and newly assigned by pressing "Apply".

Pattern and number of nodes

This option has to be selected if a pattern and subdivision data for individual elements should be assigned.

Pattern ID: The pattern ID must be given. Available patterns are shown in the following chapter and can be selected out of a list box (not all available pattern IDs are shown in the list box). If you don't specify an ID, currently defined IDs for the elements remain.

Number of intermediate nodes: Give the number of intermediate nodes for the edges of the bottom surface of the element in counter clockwise order. With elements of type 150 or 400 the ratio of largest to smallest FE edge length should not be greater than 4.

Length ratio: For some pattern it's possible to specify the ratio between the last and first segment length for directions 1 and 2.

Change pattern ID

Using this option the assigned pattern ID may be changed for the selected elements without changing the number of intermediate nodes for the edges of the elements. The old ID and the new ID must be given. For example the ID 142 may be changed to ID 150.

3. Direction of solid elements

This option has to be used if the subdivision data for the 3. direction of solid elements should be assigned. For parameter m the number of intermediate nodes must be given, optionally also the ratio between the last and first segment length (parameter q) can be given.

Element selection

Select elements the pattern should be assigned to.

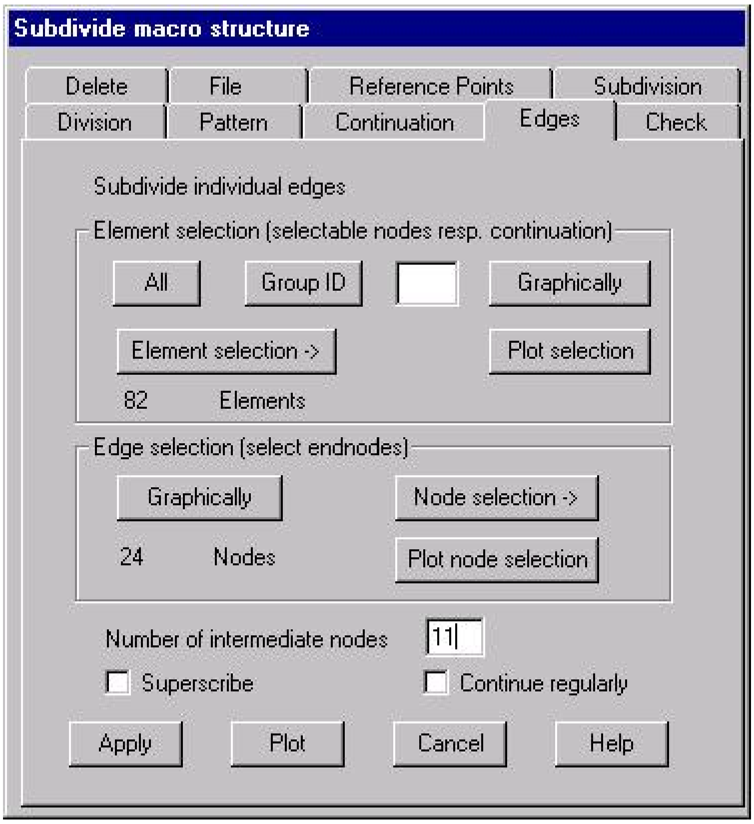
Continuation: Continue with regular subdivision

This command will apply existing subdivision data to opposite edges and adjacent elements, which have no specified subdivision data. This will continue until an edge is found with already defined data or until a boundary edge is detected. The continuation will happen only for selected elements. In case an edge with inconsistent subdivision data is detected, the data won't be corrected automatically. This can be done with the command **Check.** Regular continuation across elements of type 105 or 400 isn't possible.

Edge: Assign number of intermediate nodes to individual edges

This command specifies the subdivision for individual edges.

Following dialog shows the available options:



Element selection

Only nodes contained in the given element selection are graphically selectable. Eventually continuation uses this element selection.

Edge selection

All end nodes of the edges have to be selected that shall be assigned the given number of intermediate nodes. For solid elements only nodes of the bottom surface are selectable.

Number of intermediate nodes

Specify the desired number of intermediate nodes for selected edges.

Overwrite

If this option is not marked, edges that have already a number of intermediate nodes assigned to are not changed.

Continue regularly

With this option set a regular continuation for all adjacent edges of elements will be performed as with command **Continuation**.

command **Continuation**.

Reference Point: Define reference points for division of macro elements

This command allows defining reference points on edges of macro elements that shall become finite element nodes when the macro element is subdivided. The reference points are assigned to edges of macro elements; the corner nodes of these edges must graphically be selected. For each edge with reference points, the number of those points, their relative position and the number of intermediate nodes that should be generated between these points is stored. A maximum of four reference points may be given for one edge. The number of intermediate nodes between the reference points may also be defined using the command **Division** where each part of edges with reference points is divided individually.

The definition of reference points is saved within the file project.mut.

In most cases following steps may be useful (see demo “referencepoint.dem”).

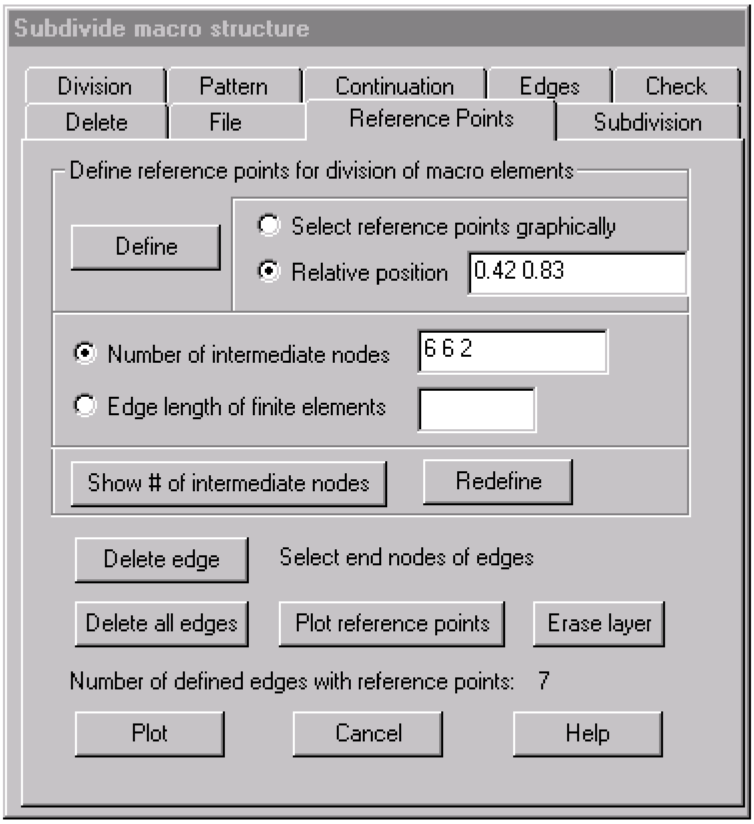
1) Define the position of reference points, without defining the number of intermediate nodes.

2) Divide all edges using the command **Division**.

3) Do a graphical test of the subdivision of some elements using command **Subdivision** and the option “Test individual elements”.

4) If there are triangular finite elements are generated for some macro elements, correct the number of intermediate nodes for some parts of edges with reference points by using the option “Show # of intermediate nodes / Redefine”.

Following dialog shows the available options:



Define reference points for division of macro elements

<u>*Select reference points graphically*</u>: If this option is marked, after selecting the end nodes of an element edge the reference nodes on that edge have to be selected graphically. Selection of reference nodes is finished with the right mouse button, then the next edge may be selected or finish selection with right mouse button. The reference points to be selected must lie within a small tolerance on the edge.

<u>*Relative position*</u>: Using this option, the relative position ($0 < s < 1$) for up to 4 reference points must be given in the corresponding input field.

<u>*Number of intermediate nodes*</u>: Using this option, for each part of the edge the number of intermediate nodes that should be generated when the edge is subdivided must be given in the input field. Missing values are set to 0. The number of intermediate nodes may be omitted when it is later determined by using command **Division**.

Edge length of finite elements: Using this option, the maximal length of finite element edges must be given in the input field. The parts of the edge are then subdivided as with command **Division**.

Define: After clicking this button, edges with reference points have to be selected graphically by selecting the end nodes of the edge. Selection is ended with the right mouse button, eventually also the reference nodes on the edges must also be selected graphically if option “select reference points graphically” is marked. If there are already reference points defined for an edge, these are replaced by the new definition.

Show number of intermediate nodes / Redefine: Clicking the first button, an edge must be selected graphically by selecting the end nodes, then the relative position of reference points on this edge and the number of intermediate nodes for the parts of the edge is shown in the corresponding input fields. After that, the number of intermediate nodes may be altered in the input field and be saved for the latest selected edge by clicking the button “Redefine”.

Delete edge

After clicking this button, the end nodes of the edges must be graphically selected that reference points should be deleted. Selection is ended with the right mouse button.

Delete all edges

Clicking this button, all defined reference points are deleted.

Plot reference points

Clicking this button, all edges that have reference points are plotted together with the reference points and the intermediate nodes on the parts of the edge.

Erase layer

Clicking this button, the layers on which the reference points and the intermediate nodes are plotted on are erased.

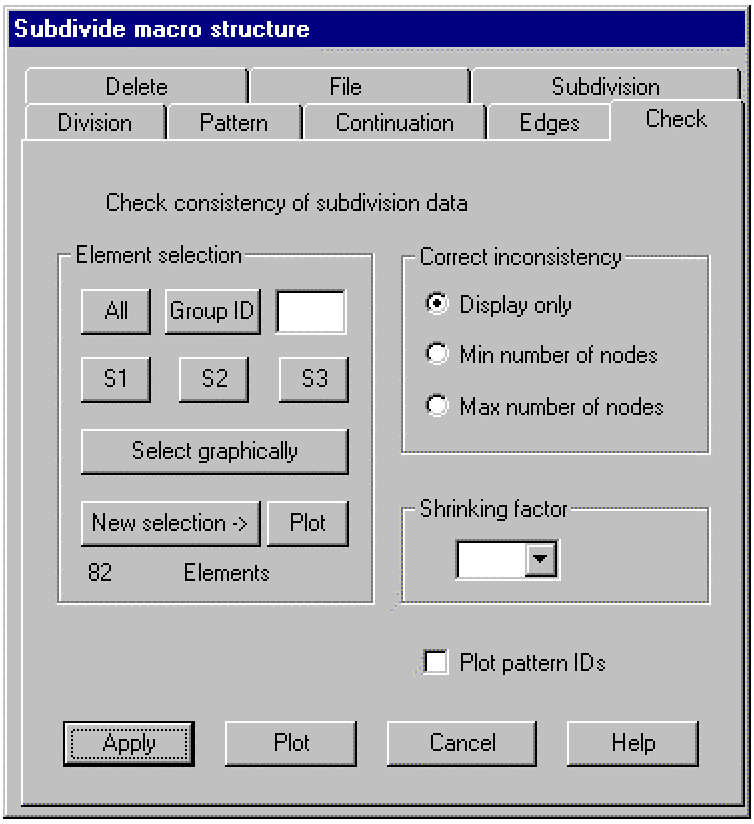
Plot

Clicking this button, the intermediate nodes on all element edges are plotted.

Check: Check consistency of subdivision data

All currently defined data for subdivision of macro elements will be checked for consistency between adjacent elements and edges. Adjacent elements or edges only can be detected if they own common end nodes. Edges with inconsistent data found are graphically marked. With solid elements, only edges of the button surface are checked.

Following dialog shows the available options:



Element selection

Select elements that are to be considered.

Correct inconsistency

Display only: Inconsistencies found are shown graphically but not corrected.

Min number of nodes: Correct the subdivision of edges by applying the minimal number of intermediate nodes for the subdivisions of all adjacent elements. This can also be zero, if no subdivision for one edge is given.

Max number of nodes: Correct automatically by applying the maximal number of intermediate nodes for the subdivisions of all adjacent elements.

Shrinking factor

For plotting purposes a shrinking factor can be given here. This makes it possible to better show inconsistencies between neighbored edges.

Plot pattern ID

With this option set all macro elements with irregular patterns will show the associated pattern ID in the center of gravity when plotted.

Delete: Delete subdivision data

The currently assigned data for subdivision will be deleted for all or selected elements. An element set must be specified in the dialog.

File: Save/load subdivision data to/from file

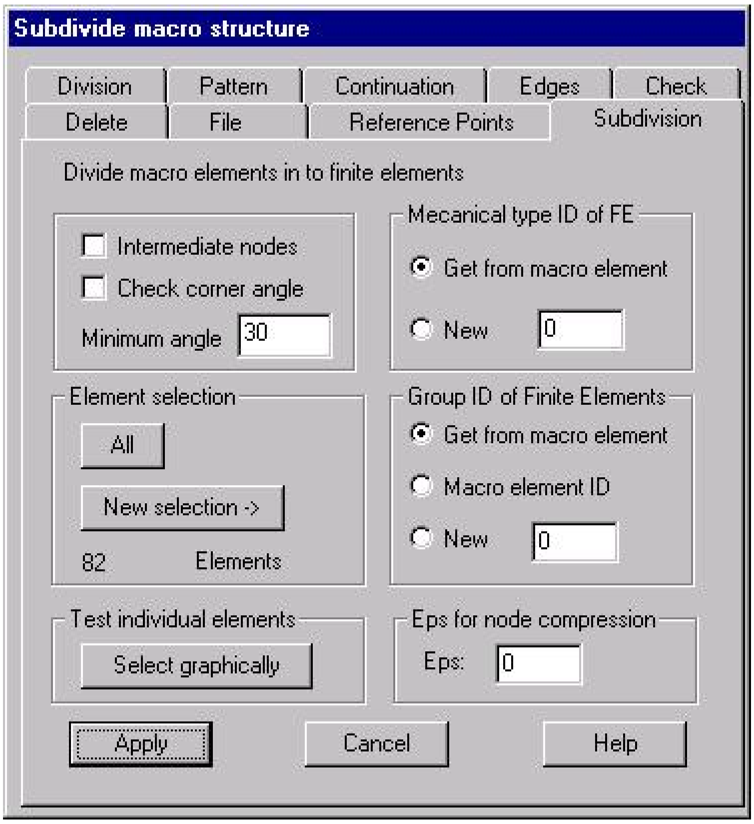
Button „Save“ saves to binary file with extension .mut. By pressing „Load“ a previously saved binary file can be loaded. The association between subdivision data and elements is done by the external element IDs, that means the IDs for elements shouldn’t be modified after saving subdivision data.

The dialog shows the current project title. Selecting options „Project title“ or „Another title“ the project title or a different title is used for saving. Clicking button “Another file title” pops up a file selection box where a file title is to be selected.

Subdivision: Perform subdivision

After specifying all necessary subdivision data, this command performs subdivision by generating finite elements.

Following dialog shows the available options:



Intermediate nodes

If this option is marked, one intermediate node is generated on the edges of the finite elements. Except for elements of type 150 or 140 this is done when the pattern for the unit element is calculated. Transferring this pattern to the geometry of the macro element results in curved edges for the finite elements. With type 150 or type 400 elements all edges of finite elements are straight.

Check corner angle

If this option is marked, all corner angles of quadrilateral elements are checked. If the angle is less than the given angle or greater than 180 degree – given angle, the quadrilateral element is replaced by two triangular elements.

Element selection

By specification of an element set it's possible to subdivide only parts of the current macro model. This especially will be useful for subdividing individual parts or for trying out the influence of different parameters for individual elements. By default all elements will be subdivided.

Group ID of finite elements

Get from macro element: Group ID of the related macro element will be adopted by the finite elements

Macro element ID: Finite elements will have a group ID the same as the original macro element ID.

New: Given ID will be assigned to the finite elements.

Mechanical type ID of finite elements

Get from macro element: Type IDs of the original macro elements are used for the resulting finite elements.

New: Given ID will be assigned to the finite elements.

Eps for node compression

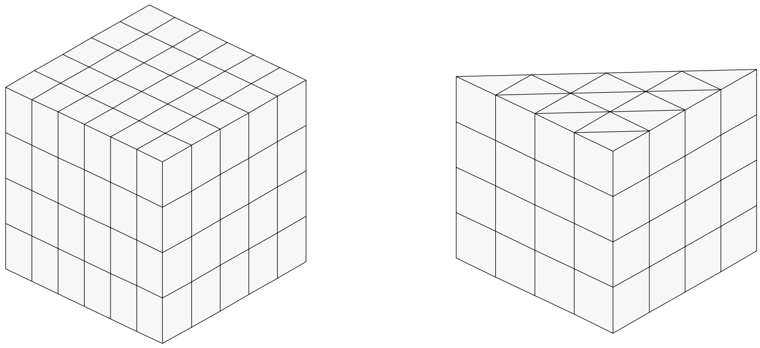
You may define a tolerance value for combining nodes within a near distance into a single node. Omitting this value will cause the program to determine a suitable default. All nodes with a distance smaller than this value are combined into a single node. Attention: If you do not give a value for eps the chosen value might be to small. In any case check whether there are nodes within a small distance that should be merged into a single node by plotting sharp edges are by using the command **Compress nodes**.

Test individual elements

Select graphically: After clicking this button macro elements have to be selected graphically, these elements will be subdivided immediately and generated finite elements are plotted using a different color, so the proper matching of the select subdivision pattern can be checked graphically.

Subdivision pattern

Each subdivision of triangular or quadrilateral macro elements is based on a pattern. Regular patterns subdivide opposite edges with the same number of segments, as is shown in the following figure:



For irregular patterns at least 2 opposite edges have a different number of segments. This way a mesh refinement is achieved into one or two directions. A 2 digit ID distinguishes the patterns for regular and a 3 digit ID for irregular subdivisions. The pattern ID for regular subdivisions is 20 for macro elements 20-25, 30 for macro elements 30-35,60-65 and 40 for macro elements 40-45,80-85.

Besides the pattern ID the number of intermediate nodes for the edges of the macro element must be given.

With some patterns also a quotient q for the length of last to first FE edge can be given. The length of FE edges increases linearly for $q > 1$ and decreases linearly for $q < 1$.

In case of solid elements a pattern is always applied to the bottom and the top surface. The direction perpendicular to the bottom surface will always be regularly subdivided, for this direction, the number of intermediate nodes and optionally a quotient for last to first FE edge length must be given.

Following m1,m2,m3,m4 are the number of intermediate nodes for edges 1-4 of a quadrilateral macro element, where the edges are numbered counter clock wise beginning with the first element node. Direction R1 points from first to second node and direction R2 from second to third node.

With a regular pattern only values for m1 and m2 must be given, m3 = m1 and m4 = m2 are set automatically. Also quotients q1 and q2 for the two main directions of a quadrilateral macro element may be given.

With an irregular pattern the last two digits of the pattern ID define the subdivision pattern as shown in the following pictures. The first digit of the pattern ID determines the rotation of the pattern. The pattern is first applied to an unit element and is then mapped to the geometry of the macro element, where the first digit of the pattern ID gives the node of the macro element that coincides to the first node of the unit element.

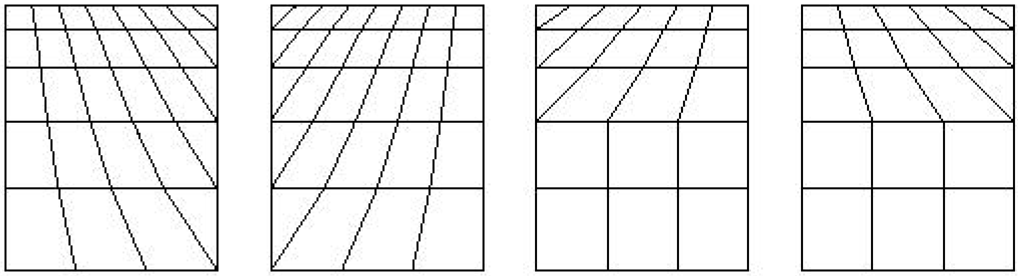
Following pictures show the different subdivision patterns.

Pattern 101, 102, 104, 105

With pattern 101 and 102 a linear increase of the number of intermediate nodes is applied for direction R1 towards direction R2. The number of intermediate nodes for edge 1 (m1) and edge 2 (m2) must be given, m3 = m1+m2+1 and m4 = m2 are set.

With patterns 104 and 105 also m3 must be given, with abs(m3-m1) < m2. The smaller number of intermediate nodes will at first be constant in the direction of R2 and then linearly increased.

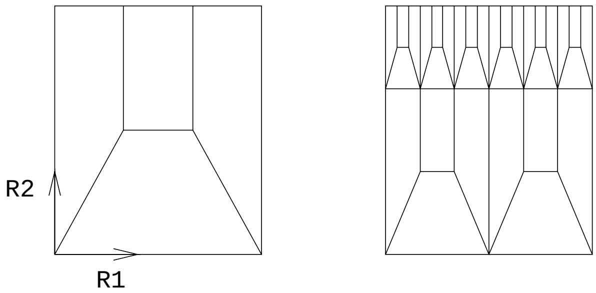
For direction R2 a quotient q for last to first FE edge length may be given.



Pattern 121

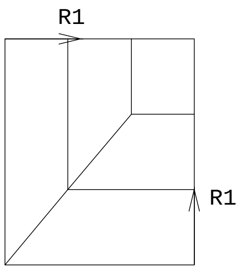
A base pattern which subdivides a quadrilateral element into 4 FE elements as shown in the left following picture is lined up in direction R1 and R2. The number of intermediate nodes m1 and m2 for direction R1 and R2 must be given, m4 = m2 and m3 = (m1+1)*(3**(m2+1))-1 is set.

For direction R2 a quotient q for last to first FE edge length may be given.



Pattern 122

This pattern allows refinements in 2 directions by giving the subdivision parameters m1 and q1.

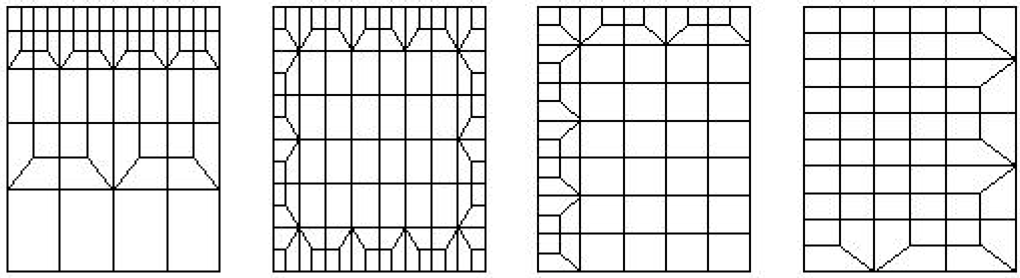


Pattern 123, 124, 125, 126

Pattern 123 does refinements in direction R2. m1,m2,m3 must be given, m4 = m3 is set, m1 must be odd, for m3 the condition m3 = (m1+1)*(2**n)-1 with n <= m2+1 must be met. For direction R2 a quotient q for last to first FE edge length may be given.

For pattern 124 m1 and m2 must be given, m3 = m1 and m4 = m2 are set.

For pattern 125 and 126 m1 and m2 must be given, m3 = 2*m1+1 and m4 = 2*m2+1 are set.



Pattern 133, 134, 135, 136

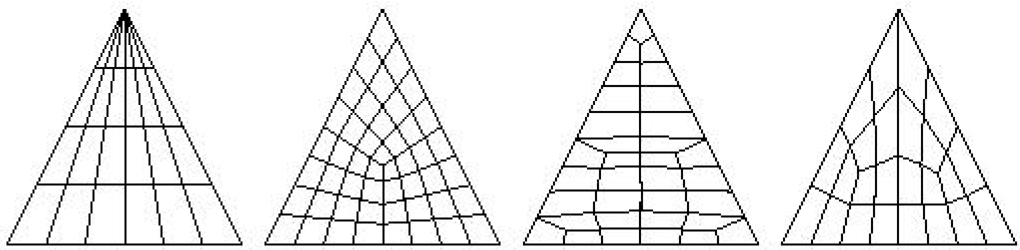
These are patterns for triangular macro elements, where m3 = m2.

With pattern 134 all edges are equally subdivided (m1 = m2 = m3) where m1 must be odd.

With Pattern 135 m1 must be odd and at least 1 and m2 must be at least 3 where m1 and m2 are independent of each other.

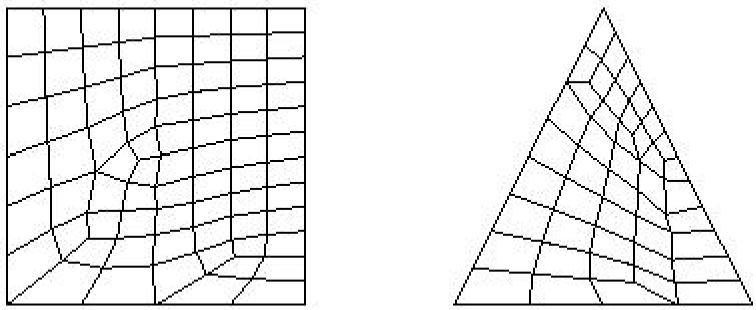
Pattern 136 is built upon 2 triangles by using pattern 134. m1 must be at least 3. Correlation between m1 and m2 is given by m1 = 2*m2+1 where m2 must be odd. Only m1 must be specified, m2, m3 are automatically set to m3 = m2 = (m1-1)/2. Pattern 136 especially may be useful when subdividing a semicircle area modeled by elements of type 32 into quadrilateral elements.

Pattern 134 is the default when subdividing macro elements of type 30-35, 60-65



Pattern 142

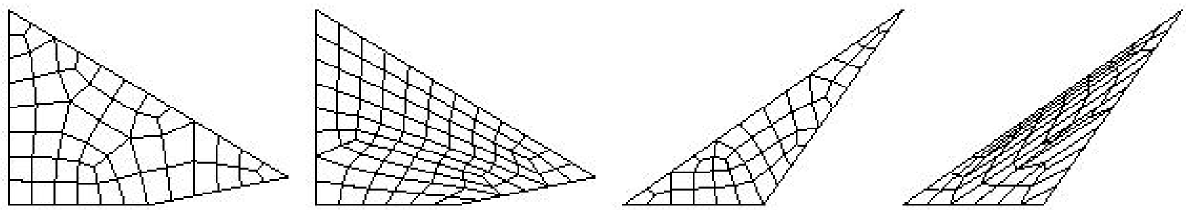
This pattern allows specifying independent numbers of intermediate nodes for all edges of the underlying macro element. If the sum of all intermediate nodes on edges is even for quadrilateral elements or odd for triangular elements, exclusively quadrilateral elements are generated. In the other cases one additional triangular element is generated.



Pattern 150 for triangular and quadrilateral macro elements with curved edges

As with pattern 142 the number of intermediate nodes may be different for all edges. But with ID 150 the geometry of the macro element is taken into account for the definition of the pattern. The macro element is projected on a plane defined by the three first corner nodes of the macro element. The triangular or quadrilateral element in this plane is than subdivided using the algorithm described below for element type 400. This net is then mapped to an unit element and this pattern is further mapped to the geometry of the given macro element. The condition must be met, that the quotient of greatest to smallest FE edge length is not greater than 4.

Pattern 150 should be used instead of pattern 142 when the angle of some corners of the macro element differ greatly from 90 degree. Following pictures show the differences of pattern 150 (left) and pattern 142 (right) for a quadrilateral and a triangular element.

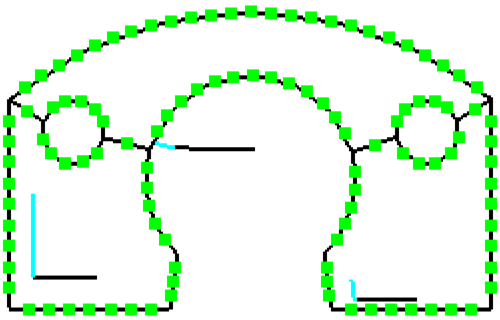
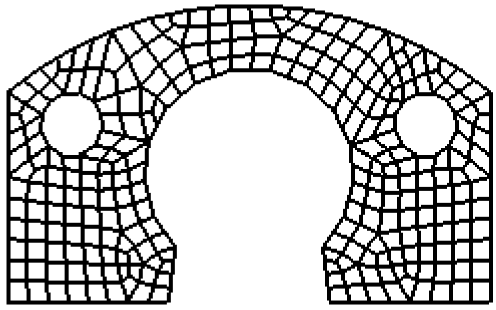


Pattern for element type 105

Macro element type 105 is a planar element with up to 10 edges consisting of straight lines, circular arcs or spline curves. For each edge the subdivision must explicitly be specified using the commands **Division** or **Pattern**. With **Pattern** the following subdivision patterns can be applied.

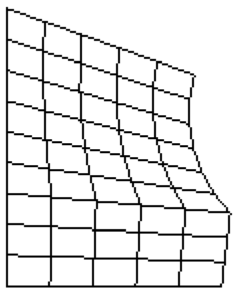
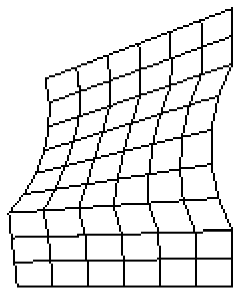
Pattern 150

This invokes a free mesh generation as with element type 400 (see below).

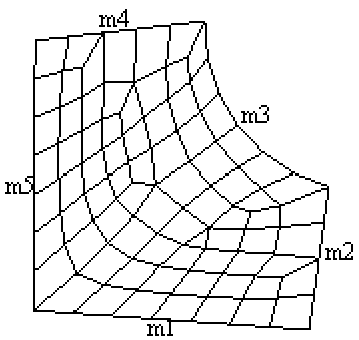
Pattern 40 or 142

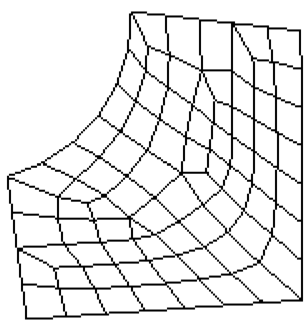
These patterns can be used if 4 vertices of the polygon can be identified as corner nodes of a quadrilateral element. The first vertex of the polygon is always the first corner of the quadrilateral element. The remaining 3 corner nodes can be specified using the command **Element definition** and will be saved after the elements height. If no corner nodes are specified these vertices of the polygon are automatically selected which give the smallest inner angles.

Pattern 151 - 551:

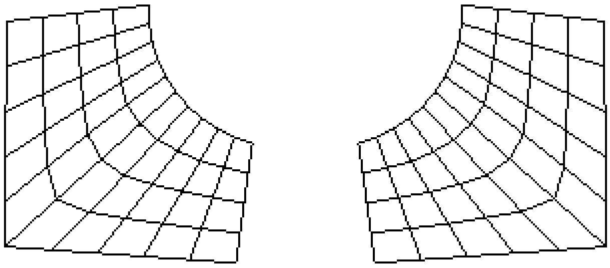
These pattern IDs can be used with macro elements of type 105 with 5 edges. At first, the element is divided into two quadrilateral elements by a line from a corner of the polygon to the middle of the opposite edge. The two resulting quadrilateral elements are divided by using the pattern 142. The first digit of the pattern ID tells, which vertex of the polygon is used for the division into two quadrilateral elements. Following picture shows pattern 151 and 251 if lower left vertex is the first element node.





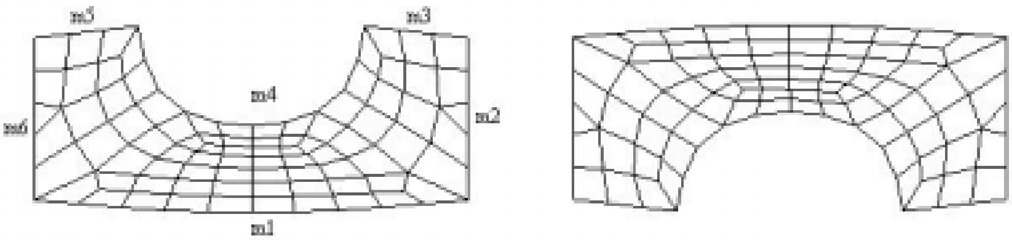
Pattern 152-552:

The pattern corresponds to pattern 151-551 with the difference that conditions for subdividing the edges 3 and 4 must be so that a regular subdivision with quadrilateral elements can be achieved. These conditions are: m4 = m2 and m3 = m1 + m5 + 1; m4 and m3 are corrected if they don't meet this condition. The first digit specifies which vertex will be the starting vertex for subdivision into 2 quadrilateral elements. Following picture shows pattern 152 and 252 (see also demo "pattern15x.dem").



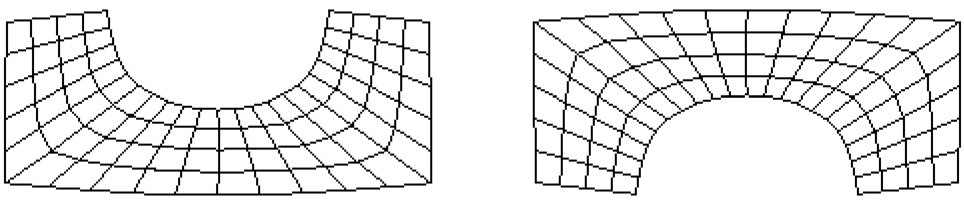
Pattern 161-661:

This pattern corresponds to pattern 151-551. It can be used for elements with 6 edges. The element is first subdivided into three quadrilateral elements, which are then subdivided by pattern 142. The first digit of the pattern ID tells which vertex of the polygon will be the starting vertex for division into three quadrilateral elements. Following picture shows pattern 161 and 561 if the lower left vertex is the first node of the element.



Pattern 162-662

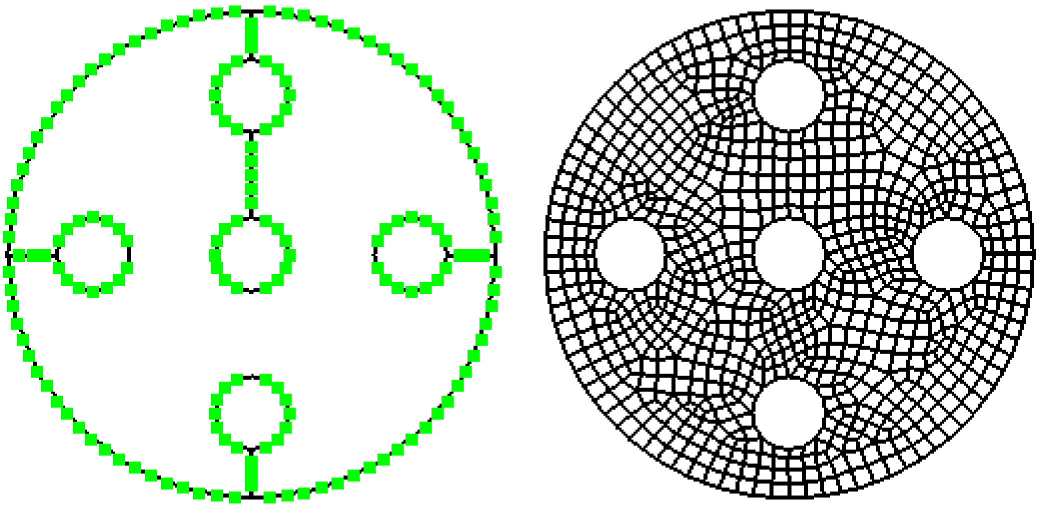
The pattern corresponds to pattern 161-661 with the exception that conditions for subdivision of edges 4 and 5 must be so, that a regular subdivision to quadrilateral elements can be achieved. These conditions are: m5 = m3 and m4 = m1 + m2 + m6 + 2; m5 and m4 are automatically corrected if they do not meet these conditions. Following picture shows pattern 162 and 562.



Subdivision of type 400 elements, pattern 150

Macro element type 400 is a planar element with up to 39 edges consisting of straight or circular lines. For subdivision of the element into finite elements, the edge length of the finite elements to be generated must be given using command **Division**. First the surrounded border of the element is subdivided using this value. Beginning at the borders, quadrilateral and triangular elements are created until the whole area is meshed. If the number of intermediate nodes on all edges is odd, subdivision is done using the double edge length, then all quadrilateral elements are subdivided into 4 and all triangular elements are subdivided into 3 quadrilateral elements; so one gets only quadrilateral elements. This first subdivision with double edge length is only applied, if strongly curved edges have at least 3 intermediate nodes.

Using command **Pattern** it is also possible to give the number of intermediate nodes for all macro element edges in counter clock wise direction, but the quotient of largest to smallest FE edge length must not be greater than 4. The whole area is meshed using a FE edge length that is equal to the mean value of the length given for all macro element edges.



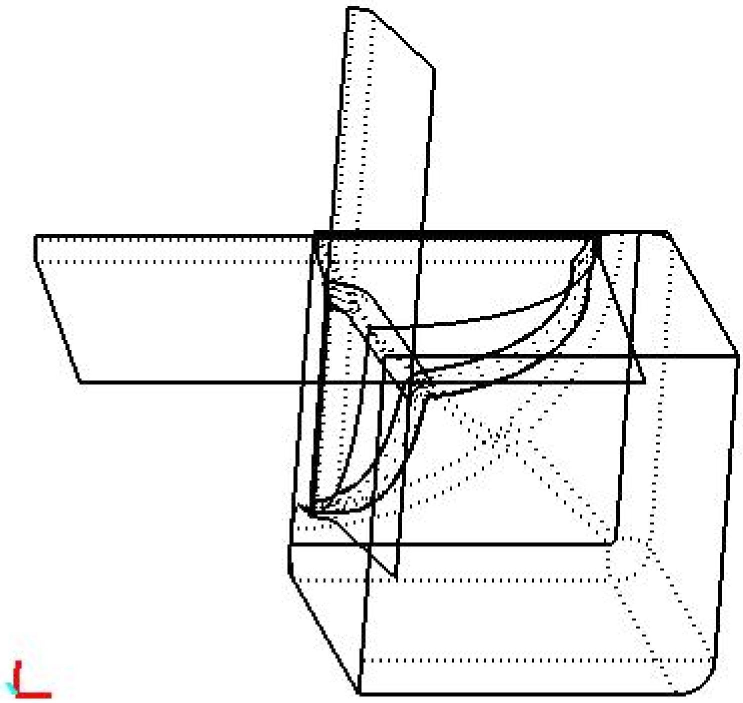
Handling CAD data in VDAFS format

VDA surface interface (VDAFS)

VDAFS is an interface to exchange data between different CAD systems or to exchange data between CAD systems and other applications, where the data exchange of free surfaces and bounded surfaces is especially important.

VDAFS is developed by the Association of Automobile Industry (VDA). For a detailed documentation of the interface see DIN 66301.

Following picture shows a CAD model of a container built out of 8 bounded surfaces (see demo file vdafs.vda) .



Using MAKROS-A following types of VDAFS elements can be read from a VDAFS input file and used to generate a finite element model:

POINT: (x,y,z)- coordinates of single points

PSET: sequence of single points

CIRCLE: arc of a circle

CURVE: curve in 3D space

CONS: curve on a free surface

SURF: free surface in 3D space

FACE:bounded surface

MAKROS-A especially provides methods to generate interactive macro elements on bounded surfaces (see demo file “vdafs.dem”).

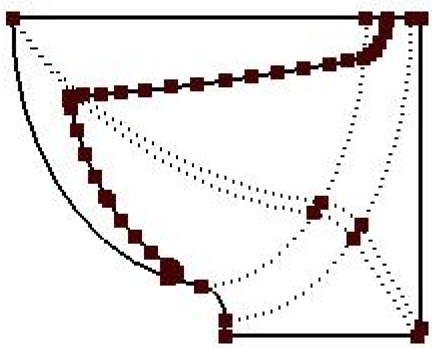
Handling of different VDAFS element types with MAKROS-A

POINT, PSET, CIRCLE, CURVE

VDAFS elements of type POINT and PSET are handled in MAKROS-A as point elements (element type 1), they can be used to transfer coordinates of single points. VDAFS elements of type CIRCLE and CURVE are approximated by one or more macro elements of type 22 (arc element).

FACE and SURF

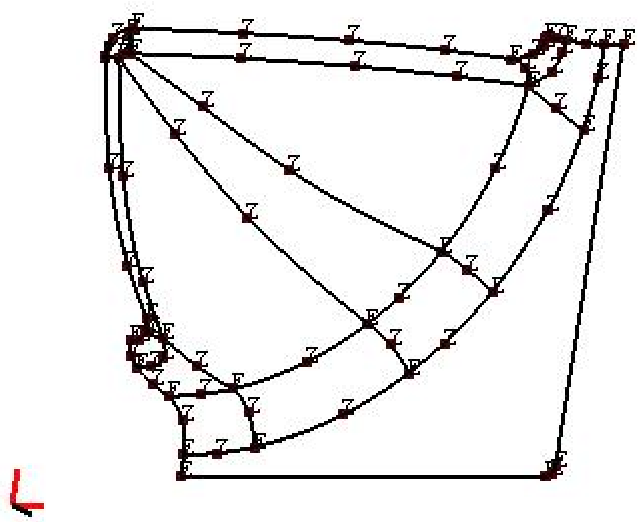
Bounded surfaces are defined in VDAFS as free surfaces (SURF element) wit one or more closed curves (CONS elements) on the surfaces. SURF elements and CONS elements mostly consist of more than one segment. Plotting SURF elements, inner edges of segments are shown as dotted lines. Following picture shows one SURF and CONS element together with automatically generated nodes at the corners of SURF segments and CONS segments:



The generation of macro elements on one bounded surface is done in MAKROS-A by the following steps:

1) Generate nodes on the surface. There are different means to do this (see dialog "**Node generation**").

2) Define macro elements using these nodes (see dialog "**Element definition**"). It is important that macro elements of neighbouring surfaces use on the common edge nodes that have nearly identical coordinates. Generating the entire macro model, these nodes are merged into one single node. To identify nodes of macro elements of neighbouring surfaces, corner nodes of macro elements are marked by the symbol "E" and intermediate nodes on the edges of macro elements are marked by the symbol "Z". Following picture shows the macro elements of two neighbouring surfaces where the defining nodes are marked "E" or "Z" respectively:



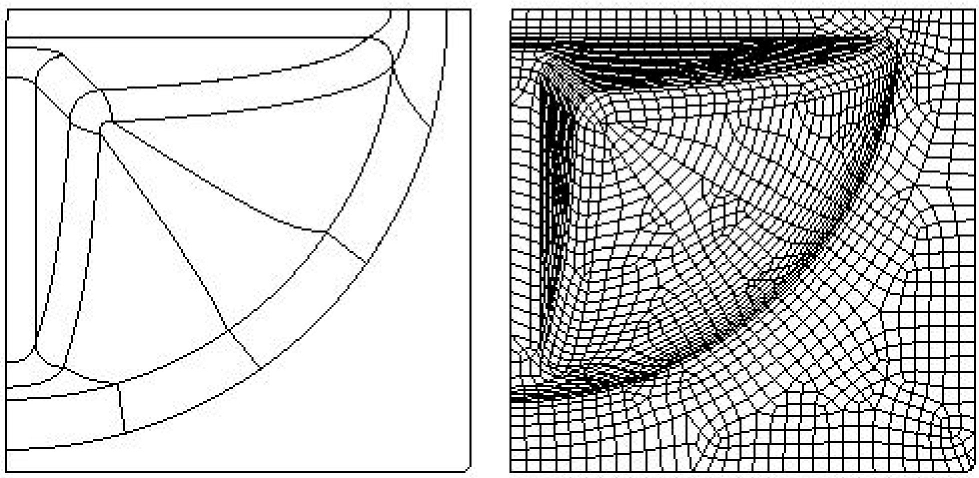
3) Generate a macro model (see dialog "**Macro- / FE structure**"). After all VDAFS surfaces have been approximated by macro elements, the entire macro model is generated using the macro elements of all surfaces,

where the nodes on the edges of neighbouring surfaces are merged into one node, if they have equal coordinates within a given tolerance.

4) Subdivide the macro elements into finite elements. This is done as with an ordinary macro model (see chapter "Subdivision of macro elements into finite elements").

5) Smooth the finite element model if necessary (see dialog "**Macro- / FE structure**"). The calculation of finite element nodes when subdividing the macro elements is done by C0 – Coons interpolation between opposite edges of the macro element. Depending on how large and strongly curved the macro elements are, the generated nodes will differ from the geometry of the originating SURF element. This difference may be eliminated by smoothing the finite element model. The following algorithm is used: For each macro element an optimal plane is determined, where the nodes of the finite elements belonging to the macro element are projected. Then, nodes on the originating SURF element are calculated, that have the same projection on the plane. These nodes on the SURF element replace the finite element nodes calculated by the Coons interpolation. To determine an optimal plane, no small macro elements must be used, also the macro elements must not be to strongly curved.

Following picture shows the generated macro model and final finite element model of the container shown above (see demo file "vdafs.dem").



Dialogs to handle VDAFS data

The dialogs to handle VDAFS data are grouped together in a window using the following pages.

File	Read VDAFS input file, save data to or load from a binary file
Plot	Plot VDAFS elements
Node generation	Generate nodes on bounded surfaces
Element Definition	Define macro elements on SURF elements
Macro- / FE structure	Generate a macro and FE model, smooth the FE model
Select / Remove	Define a VDAFS element selection, remove VDAFS elements

The window is activated by the command "**VDAFS data**" in the menu group "**Generation**" and deactivated using button "*Cancel*".

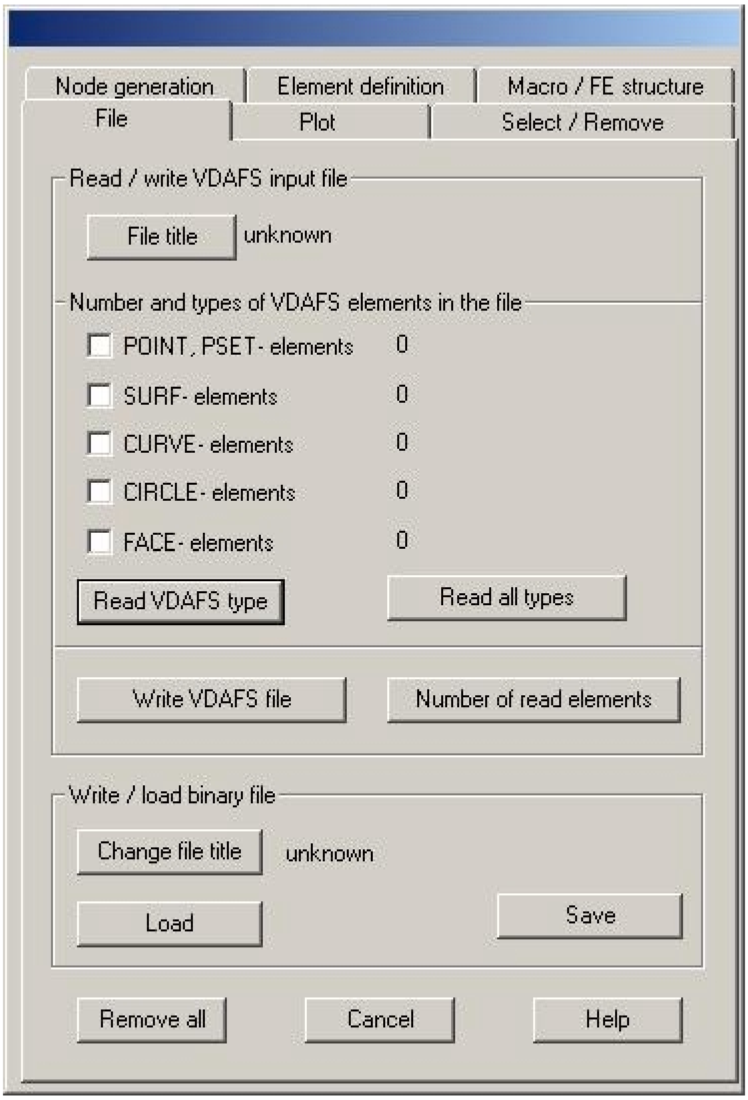
As long as the window is active all other commands to modify the macro or finite element model are disabled.

Attention: Before a new VDAFS input file is read or a VDAFS binary file is loaded, all data that is actually in memory (macro model, FE model, VDAFS data etc.) is removed and must eventually be saved to disk. If several VDAFS files shall be combined in one finite element model, first one macro and FE model must be generated for each VDAFS file. Then, these structures can be combined in one structure using command "**Load structure**".

„File" dialog

This dialog is used to read a new VDAFS input file or to save VDAFS data together with generated macro elements to a binary file or to load a binary file into memory. VDAFS input files must have the extension ".vda" and binary files must have the extension ".vd".

Following dialog shows the available options:



Read / Write VDAFS-ASCII file

File title: Clicking this button, a file selection box pops up where the title of the input file is to be selected. After selecting an input file, the number of different types of VDAFS elements in the file is determined and shown in the dialog.

Number and types of VDAFS elements in the file: If not all different element types of the file are to be read, the types to be read must be selected.

Read VDAFS type: Clicking this button, only the marked types of VDAFS elements are read.

Read all types: Clicking this button, all types and elements of the input file are read.

Write VDAFS file: Clicking this button, a new VDAFS-ASCII file is created. The file title must be given in a file selection box. Only VDAFS elements that are contained in the active VDAFS selection (see dialog "**Select / Remove**") are written to the output file.

Number of read elements: Clicking this button, the number of elements of the different VDAFS types in memory is listed in the protocol window.

Write / Load binary file

Change file title: Clicking this button, a file selection box pops up where the title of the new binary file must be given, the extension ".vd" is added automatically.

Save: Clicking this button, the actually VDAFS data in memory, together with nodes and macro elements assigned to SURF elements, are written to the given file.

Load: Using this button, a binary VDAFS file is loaded into memory, all existing data in memory is removed before.

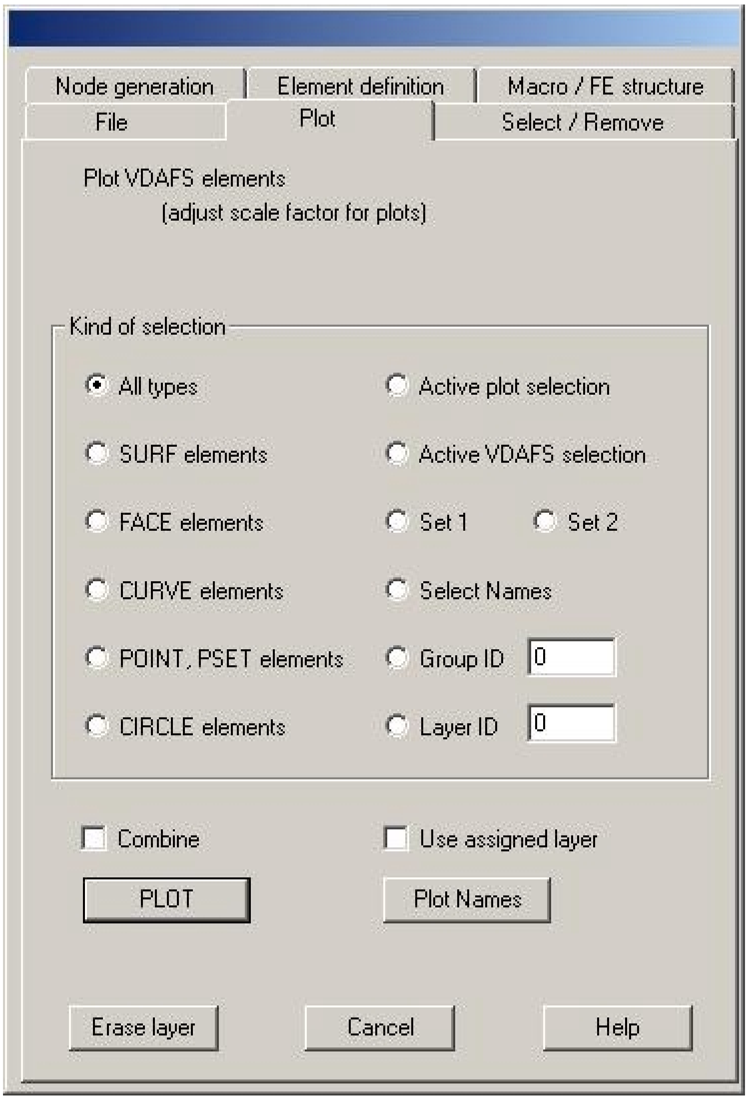
Remove all

Clicking this button, all data in memory is removed.

"Plot" dialog

This dialog enables to plot all or selected VDAFS elements or element types. The plotted elements define the "*active plot selection*" which is used to calculate the scale factor for plotting elements and for graphically selection of elements. If option "*combine*" is marked, newly selected elements are added to the active plot selection.

Following dialog shows the available options:



Selection

A type of selection must be marked. Options "*Active selection*", "*Set 1*" and "*Set 2*", mean all elements that are contained in this set (see dialog "**Selection / Remove**"). Using option "*Select name*", a text window with the names of all stored VDAFS elements pops up when clicking button "*Plot*". In this window, the names of VDAFS elements to be plotted must be marked with the cursor. After closing the text window, using button "OK", the marked VDAFS elements are added to the active plot selection.

Plot: Clicking this button, all elements of the active plot selection are displayed in the graphics window.

Plot names: Clicking this button, the names of all elements of the active plot selection are displayed in the graphics window.

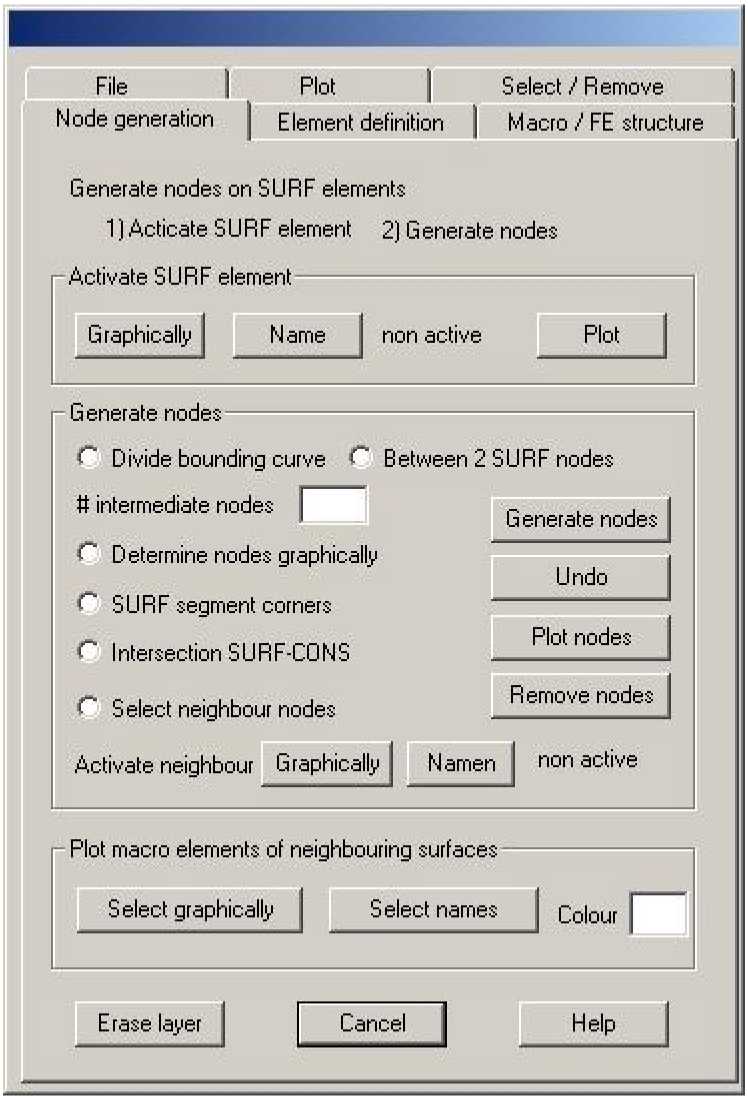
Combine: If this option is marked, selected elements are added to the active plot selection, otherwise a new active plot selection is defined.

Use assigned layer: Using dialog "**Select / Remove**", VDAFS elements may be assigned to different layers (display lists). If this option is marked, these layers are used. This allows hiding or quickly showing different parts of the structure using command "**Layer**".

"Define nodes" dialog

Using this dialog, nodes for the definition of macro elements are created. Nodes are assigned to SURF elements, so one SURF element must always be active. If a boundary curve exists for the active SURF element, this is plotted using a different colour.

Following dialog shows the available options:



Activate SURF element

<u>Graphically</u>: Clicking this button, the names of the actually displayed SURF elements are plotted with an additional symbol. Use the cursor to select the symbol of the SURF element to be activated. The newly activated SURF element is displayed together with the bounding CONS elements.

<u>Name</u>: Clicking this button, a text window pops up with the names of all SURF elements. Mark the name of the SURF element to be activated with cursor and click the OK button of the text window.

<u>Plot</u>: Clicking this button, the active SURF element and the bounding CONS elements are newly plotted.

Generate nodes

Following options may be used to generate new nodes on the active SURF element:

<u>Divide bounding curve</u>: Using this option, nodes are generated on the bounding curve of the SURF element. Nodes are generated at the end points of all CONS segments. If the number of intermediate

nodes given is greater than 1, this number of nodes is generated additionally in the inner of the CONS segments.

Between 2 SURF nodes: With this option, two existing nodes of the active SURF element are repeatedly selected graphically; then, on the line between these two nodes, m intermediate nodes are generated, where m is the number given in the input field.

Number of intermediate nodes: In the input field the number of intermediate nodes to be used with the two options above must be given.

Determine intermediate nodes graphically: With this option, you first select graphically two existing nodes of the active SURF element. A line between these two nodes is plotted, then, using left mouse button, you repeatedly mark the location of the new nodes to be generated on this line. Clicking right mouse button finishes the generation of new nodes on the active line. Then, the end nodes of a new line may be selected until the right mouse button is pressed twice.

SURF segment corners: With this option, one node is generated at the corners of all SURF segments.

Intersection SURF-CONS: With this option, one node is generated at the intersection of the edges of SURF segments and the bounding curves of the surface (CONS elements).

Select neighbour nodes: This option enables generating nodes on the bounding curve of the active SURF element that have nearly the same coordinates as existing nodes of neighbour SURF elements. First a neighbour SURF element must be activated by clicking the button “*Graphically*” or “*Name*”; all defined macro elements of this SURF element are plotted, where the nodes of these elements are marked by the symbol “E” for corner nodes and “Z” for intermediate nodes on edges. These nodes then are graphically selectable. Selecting one of these nodes by using left mouse button, a new node on the bounding curve of the active SURF element is generated, with coordinates as closely as possibly as the selected node. This requires the distance of the selected node and the bounding curve of the active SURF element to be less than a given tolerance.

Graphically: Click this button to activate a neighbour SURF element graphically.

Name: Click this button to activate a neighbour SURF element by name.

Generate nodes: After having chosen how to generate new nodes, click this button to start the operation. Newly generated nodes are immediately plotted.

Undo: Clicking this button, the latest generated nodes are removed.

Plot nodes: Clicking this button, all nodes of the active SURF element are newly plotted.

Remove nodes: Clicking this button, all nodes of the active SURF element, which have not been used for the definition of macro elements, are removed.

Plot macro elements of neighbouring surfaces

Select graphically: After clicking this button select graphically all neighbouring SURF elements of which the macro elements should be plotted. Nodes of the macro elements are marked by the symbols “E” for corner nodes and “Z” for intermediate nodes on edges. The colour index to plot the macro elements must be given. Symbols are plotted using font index 2.

Select names: Clicking this button, neighbouring SURF elements are selected by name in a text window.

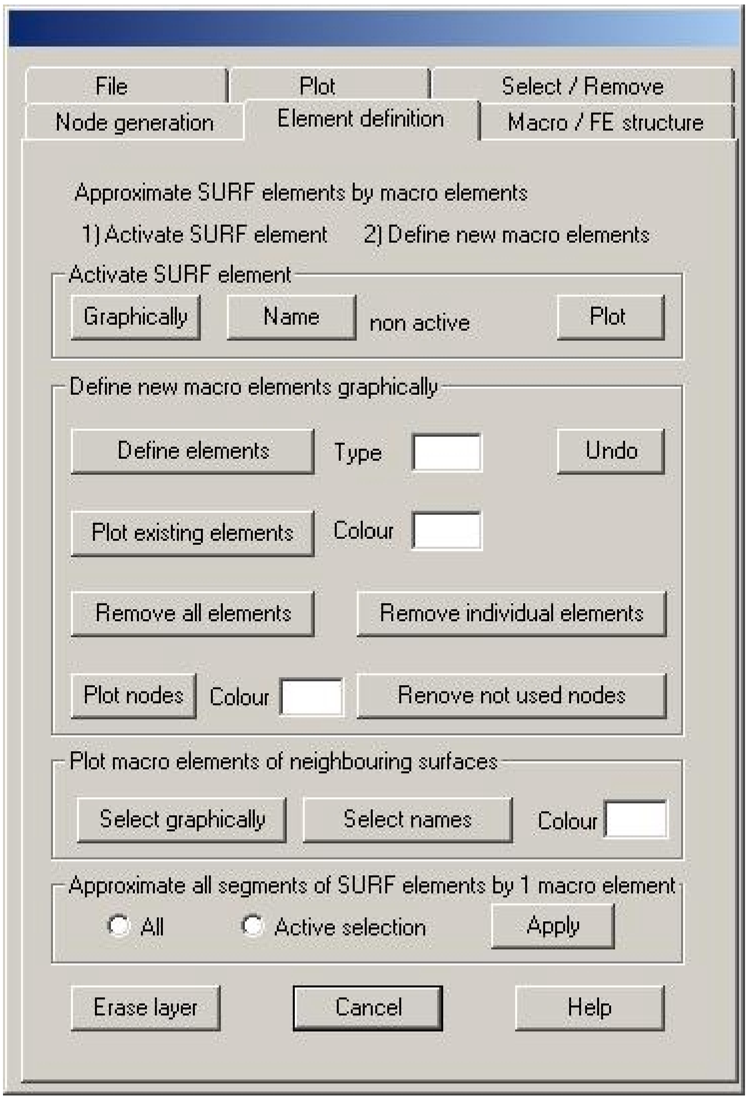
“Element definition” dialog

Use this dialog to define macro elements graphically. The nodes to be used must have been created before. It can freely be switched between the dialogs “**Node definition**” and “**Element definition**”. Newly defined macro elements are assigned to the active SURF element.

After clicking the button “*Define elements*”, the nodes of the new elements must be selected graphically. The number of nodes to be selected depends on the type of macro element. For element types with a variable number of nodes (types 35, 45, 105, 400), node selection is finished by clicking the right mouse button. For the selection of nodes the left mouse button is used, if the “–“ key is pressed before, the ID of the selected node will get a “–“ sign (last node of each edge of element types 35, 45 and all intermediately nodes on edges of element types 105, 400). Selected nodes are immediately marked with a colored symbol, pressing d key a wrongly selected node is removed. If with element type 35 or 45 an edge is straight, press 0 key for the intermediate node of this edge. Each newly defined macro element is plotted immediately. Macro elements may be defined repeatedly, clicking button “*Undo*”; the last defined macro element is removed. Finish the definition of new elements by pressing right mouse button.

After all macro elements of the active SURF element have been defined, nodes of the SURF element not used should be removed clicking button “*Remove not used nodes*”.

The following dialog shows the available options:



Activate SURF element

Graphically: Clicking this button, the names of the actually displayed SURF elements are plotted with an additional symbol. Using the cursor select the symbol of the SURF element to be activated. The newly activated SURF element is displayed together with the bounding CONS elements.

Name: Clicking this button, a text window pops up with the names of all SURF elements. Mark the name of the SURF element to be activated and click the “OK” button of the text window.

Plot: Clicking this button, the active SURF element and the bounding CONS elements are newly plotted.

Define new macro elements graphically

Define element: After clicking this button new macro elements are defined graphically. The type of the new macro elements must be given in the input field, possible types are 30, 32, 35, 40, 42, 45, 105, 400.

Definition of elements is done as with the command “**Element definition**” (see chapter “Commands to generate new nodes and elements”).

Undo: Clicking this button, the last defined macro element is removed.

Plot macro elements: Clicking this button all defined macro elements of the active SURF element are newly plotted, where corner nodes are marked with the symbol “E” and intermediate nodes on edges with the symbol “Z”.

Remove all macro elements: Clicking this button, all macro elements of the active SURF element are removed.

Remove individual elements: After clicking this button, macro elements to be removed must be selected graphically.

Remove not used nodes: Clicking this button, all nodes of the active SURF element that are not used for element definition are removed.

Plot macro elements of neighbouring surfaces

Select graphically: After clicking this button, select graphically all neighbouring SURF elements of which the macro elements should be plotted. Nodes of the macro elements are marked by the symbols “E” for edge nodes and “Z” for intermediate nodes on edges. The colour index to plot the macro elements must be given. Symbols are plotted using font index 2.

Select names: Clicking this button neighbouring SURF elements are selected by name in a text window.

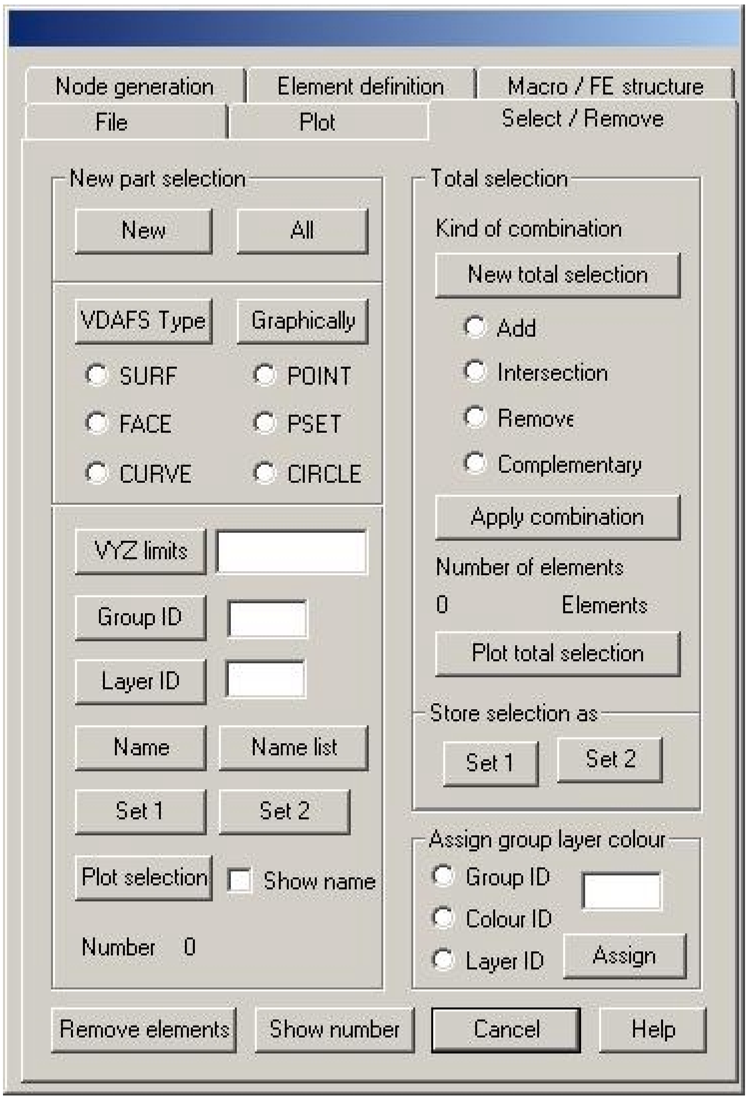
Approximate all segments of SURF elements by one macro element

Clicking button “*Apply*”, all segments of the SURF elements that are contained in the actual SURF element selection are approximated by one macro element of type 42. This option is useful when no boundary curves are given for the surfaces. To define a SURF selection, use dialog “**Selection / Remove**”.

“Selection / Remove” dialog

Use this dialog to define an actual selection of VDAFS elements that may be used by other dialogs. The actual selection is defined by combining several part selections, as is done with the general command “**Element selection**” (see chapter “General commands”).

The following dialog shows the available options:



New part selection

<u>New</u>: The actual part selection is set to zero.

<u>All</u>: All VDAFS elements are added to the part selection.

<u>VDAFS type</u>: All VDAFS elements of the selected type give the part selection.

<u>Graphically</u>: After clicking this button, VDAFS elements for the new part selection are selected graphically.

<u>XYZ limits</u>: All VDAFS elements that are fully contained within the given xyz limits give the new part selection. Xyz limits must be given as xl, xr, yl, yr, zl, zr.

<u>Group ID</u>: Selected are all VDAFS elements with the given group ID.

<u>Layer ID</u>: Selected are all VDAFS elements with the given layer ID.

<u>Names</u>: Clicking this button, the names of all VDAFS elements ordered by VDAFS types are listed in a text window. To select elements mark corresponding names with the cursor and click button “OK” of the text window.

<u>Name list</u>: Clicking this button, only the names of all VDAFS elements are shown in a text window; you cannot select names.

<u>Set 1, Set 2</u>: With thes2 buttons, VDAFS elements that are contained in the selection set 1 or set 2, respectively, give the new part selection.

Plot selection: Clicking this button, the VDAFS elements of the actual part selection are marked by a symbol and the name of the element if option "*Show names*" is marked.

Total selection

The total selection mostly is built out of several part selections. To combine the actual total selection with the actual part selection, following options may be used:

New total selection: The actual total selection is set to zero.

Add: The elements of the actual part selection are added to the actual total selection.

Intersection: Only those VDAFS elements are used for the new total selection, that are contained in the actual total selection and the part selection.

Remove: VDAFS elements of the part selection are removed from the total selection.

Complementary: All VDAFS elements that are not contained in the part selection give the new total selection.

Apply combination: Clicking this button, the combination of the old total selection with the actual part selection is executed to give a new total selection. The actual part selection then is set to zero.

Plot selection: Clicking this button, the VDAFS elements of the actual total selection are marked by a symbol and the name of the elements if option "*Show names*" is marked.

Store selection

Two different total selections may be stored as a selection set for easy reselection.

Assign group ID, layer ID, color ID

Clicking button "*Apply*" the given number in the input field is assigned as group ID, layer ID or color ID to the VDAFS elements of the actual total selection. These assignments are also used for the macro and finite elements when creating the finite element model.

Remove Elements

Clicking this button, all VDAFS elements of the actual total selection are removed in memory;

Show number of elements

Clicking this button the number of VDAFS elements of the different types is shown in the protocol window.

"Macro / FE structure" dialog

After all SURF and FACE elements have been approximated by macro elements, a macro model is created from these macro elements and the finite element model is created by subdividing the macro elements. This finite element model may then be smoothed on the basis of the originating SURF elements.

Following sequence of actions is important:

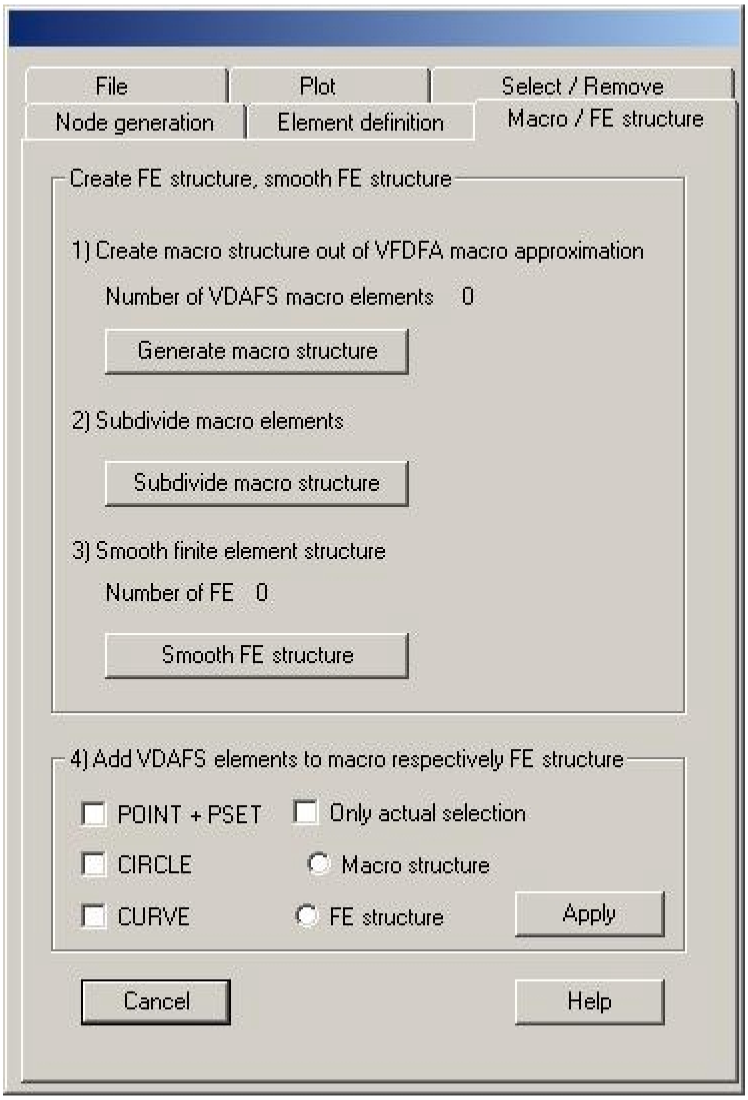
1. Generate a macro model by clicking button "*Create macro model*".

2. Generate the finite element model after clicking the button "*Subdivide macro model*". The VDAFS dialog is hidden, and the dialog for the subdivision of macro elements pops up. Using this dialog, the subdivision of the macro elements is defined and the finite element model is created as is described in chapter "Subdivision of macro elements into finite elements". After closing the dialog for macro element subdivision, the VDAFS dialog is shown again.

3. If large or strongly curved macro elements are used, it may be useful to smooth the finite element model on the basis of the originating SURF elements as described above. To do this, click button "*Smooth FE model*".

4. Add additional VDAFS element types. When creating the macro model, only macro elements that approximate the SURF and FACE elements are considered. If additional VDAFS elements of types POINT, PSET, CIRCLE or CURVE shall be added to the macro or FE model, this must be done after smoothing the FE model.

Following dialog shows the available options.



Generate macro model

Clicking this button the macro elements of all SURF elements are combined in one macro model.

Subdivide macro model

Clicking this button the VDAFS dialog is hidden, and the dialog for subdividing the macro model pops up. After the finite element model has been created, close this dialog; the VDAFS dialog will be visible again.

Smooth FE model

Clicking this button will smooth the finite element model on the basis of the originating SURF elements.

Add VDAFS elements

<u>*POINT + PSET*</u>: Single points are added as point elements (type 1).

<u>*CIRCLE*</u>: CIRCLE elements are approximated by one or two elements of type 22.

CURVE: Each segment of the curves is approximated by one element of type 22.

Only actual selection: If this option is marked, only the VDAFS elements that are contained in the actual VDAFS element selection are considered.

Macro / FE model: You have to choose whether the element shall be added to the macro model or the FE model.

Apply: Clicking this button the chosen VDAFS elements are added to the macro model or FE model, respectively.

General commands

Overview

Menu group „**General**“ contains the following general commands:

Information	General information about assignments, node coordinates, etc.
Assign IDs	Assign group ID’s, color indices, etc.
Layer names	Define name of layers
Sort nodes	Sort and renumber nodes and elements by position
Smoothing	Shift nodes on a regular surface
Compress nodes	Remove nodes which are close together
Check elements	Check conditions (vertex angle, orientation etc.) of elements
Element selection	Select elements
Node selection	Select nodes
Stop	Finish application
xunload	Exits MAKROS if invoked from AutoCAD (AutoCAD command)
MAKROS	Command selection from MAKROS menu (AutoCAD command)
Control to AutoCAD	Pass control to AutoCAD

With **Information** you get the information, which element types, groups, IDs etc. exist in the active structure. With **Assign IDs** elements are assigned new group IDs, color indices, etc., With **Layer names** layer IDs are assigned a text string, so that parts of the structure may be identified by names. With **Sort Nodes,** nodes and elements may be sorted according to their geometrical position in the structure. With **Smoothing** nodes may be shifted on to regular surfaces. **Compress nodes** serves the purpose to fuse nodes, which are close together into a single node. **Check elements** allow checking the corner angles, distortion, etc., of the elements.

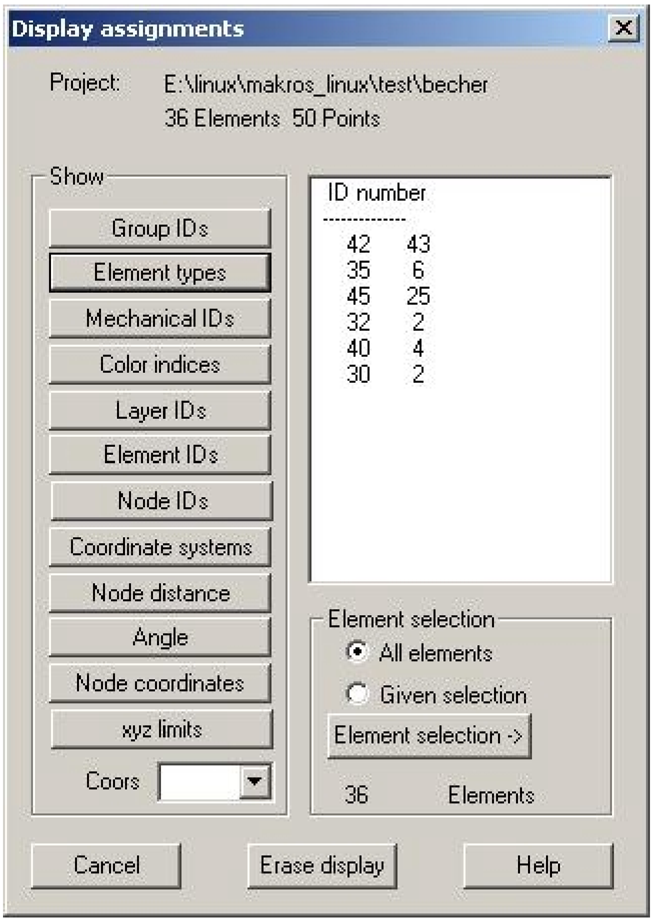
Commands **Element selection** and **Node selection** allow defining and saving of selections for the use with other command. Commands **Control to AutoCAD** and **MAKROS** are only available if the program is started from within AutoCAD, it gives control back to AutoCAD and conversely. Command **Stop,** respectively **xunload** when started from AutoCAD, ends the application.

Command description

Information: General information about the assignments, node coordinates, etc.

This command shows several kinds of information (element ID’s, group ID’s etc.) for selected elements. By pressing “Element selection” a new selection can be defined several times, from which the assignments are displayed.

Following dialog shows the available options:



Show

<u>*Group IDs, Element types, Mechanical IDs, Color indices, Layer IDs:*</u> With these buttons there will be shown the existing ID's within the structure together with the number of assignments.

<u>*Element IDs:*</u> The distribution of element ID's will be shown. First and last ID shows each continuous range of ID's.

<u>*Node ID:*</u> The distribution of node ID's will be shown. First and last ID shows each continuous range of ID's.

<u>*Coordinate systems:*</u> The ID and type of currently defined local coordinate systems are shown.

<u>*Node distance*</u>: After pressing this button 2 nodes must be repeatedly selected within the graphics window. The distance between these nodes will be calculated and shown.

<u>*Angle:*</u> After pressing this button 3 nodes must be repeatedly selected within the graphics window. The angle within the first node will be shown.

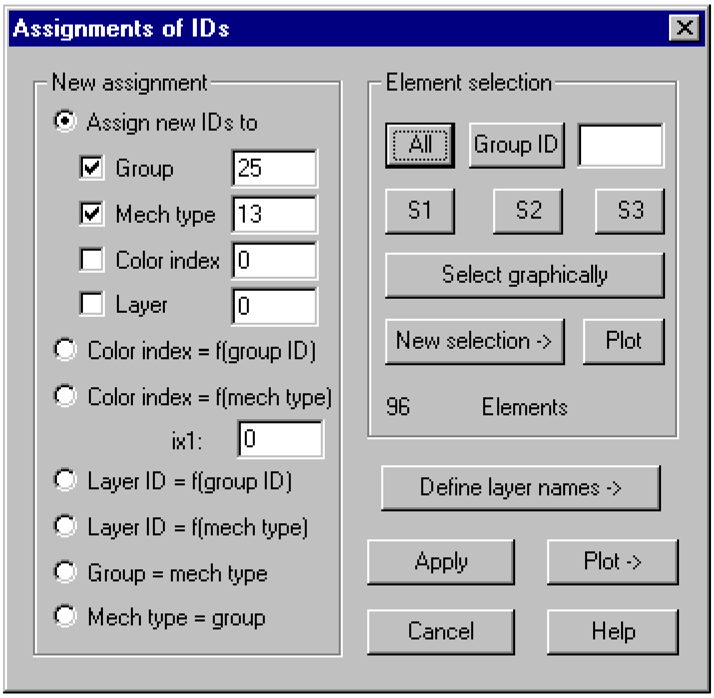
<u>*Node coordinates*</u>: After pressing this button the nodes whose coordinates should be displayed have to be selected graphically. If a local coordinate system is selected in the list box the coordinates are shown in reference to this coordinate system.

<u>*xyz-limits*</u>: Maximal and minimal x,y- and z-coordinates are shown. If a local coordinate system is selected in the list box the coordinate limits are shown in reference this coordinate system.

Assign IDs: Assign group ID's, color indices, etc.

To the elements can be assigned a group ID, mechanical type ID, color index and layer number. The command can be used multiple times for different selections. After setting the selection and the assignments button "Apply" does the assignments.

Following dialog shows the available options:



New assignment

<u>*Assign new IDs to:*</u> Assignments for group ID, mechanical type ID, color index and layer ID can be given, one or more options may be selected.

<u>*Color index = f(group ID)*</u>*:* To each element group will be assigned a unique color index so that element groups can be easily distinguished when plotting. The index of the first color must be given within the input field "ix1".

<u>*Color index = f(mech type):*</u> To each currently defined mechanical type ID will be assigned a unique color index so that mechanical type ID's can be easily distinguished when plotting. The index of the first color must be given within the input field "ix1".

<u>*Layer ID = f(group ID):*</u> Each group ID will get a different layer ID.

<u>*Layer ID = f(mech type):*</u> Each mechanical type ID will get a different layer ID.

<u>*Group ID = Mechtype ID:*</u> For the selected elements the group ID is set equally to the mechanical type ID of the elements. This may simplify the element selection; because most commands allow direct element selection by group ID.

<u>*Mech type ID = Group ID:*</u> For the selected elements the mechanical type ID is set equally to the group ID of the elements.

Element selection

Select elements that should get new IDs.

Define layer names

Pressing this button causes the dialog for command **Layer name** to be popped up which shows currently defined layer names and allows defining new names.

Plot

Pressing this button causes the command **Plot ID** to be executed to verify associations graphically.

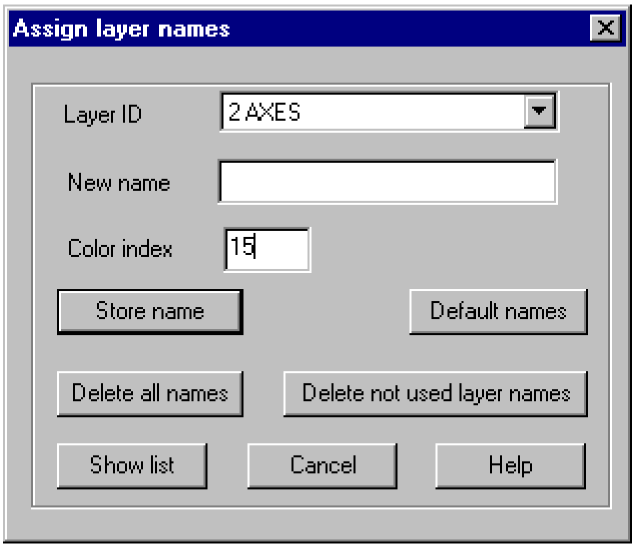
Layer name: Define layer name

To the elements can be assigned a layer ID using command **Assign Ids**. With this command a name can be assigned to the layer IDs. The use of layer names makes it possible to identify parts of the structure by name. Layer IDs and names will be saved with the structure to hard disk.

In case elements are read from AutoCAD, AutoCAD layers are continuously numbered and the IDs and names of the original layers are assigned to the elements.

To macro elements can be assigned layer IDs 1-100 and to finite elements layer IDs 101-199. When subdividing macro elements into finite elements a layer ID of original layer ID + 100 will be assigned. The layer names of original elements will be extended by the extension -fe.

Following dialog shows the available options:



Layer ID

The list box shows all available layer IDs together with their associated names. Any layer can be selected.

New name

A new layer name can be set.

Color index

Together with this layer a color index can be set.

Save names

Causes the actual settings to be saved.

Default names

All layers that have no name assigned yet will be assigned a default name in the form m_elem_xx where xx is the layer ID.

Delete all names

All layer names are deleted.

Delete unused names

In case associations between elements and layers are changed it's possible that layer names remain without associated elements. This button removes these layer names.

Show list

A list box shows all available layer names together with associated color and number of referring elements.

Sort nodes and elements: Sort and renumber nodes and elements by positions

Nodes and elements can be sorted depending on their geometrical position. The elements or nodes will at first be renumbered according to their coordinates and afterwards sorted by these numbers.

Renumbering follows these rules:

The smallest box containing all nodes is calculated.

This box is subdivided into 1024 segments for each direction, i.e. the structure is subdivided into 2^{30} small boxes. When choosing the sequence of sorting, it may be specified that this subdivision is ignored for one direction.

The boxes are sequentially numbered, for example by choosing xyz, the sequence of numbering is (((ix = 1, 1024), iy = 1, 1024), iz = 1, 1024) with ix,iy,iz the number of the section in the direction of coordinate axis.

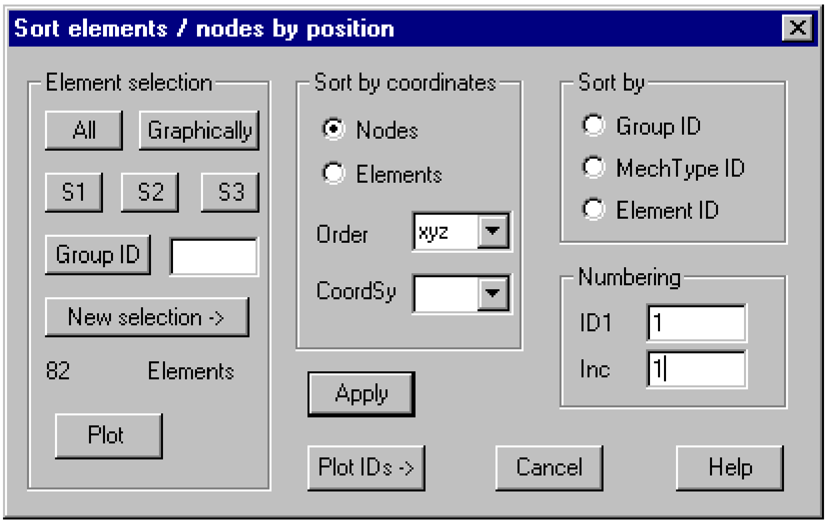
Nodes are distributed into the corresponding boxes and sequentially numbered.

In case a local coordinate system is given, all nodes are transformed into this system and sorted according to their position within this system.

When sorting elements the center of gravity is used as position of elements.

Note: If the node or element IDs of a finite element structure is changed, it is checked whether a file with load data exists and it is asked whether the IDs used in this file should be adjusted. In case of a positive answer the load data is loaded and the external node IDs respectively element IDs are changed, after that the file is saved to hard disk. In case of a negative answer, it is asked whether the renumbering should be executed anyway. Answer NO has to be given when the file has to be updated before with data in the memory. If IDs are changed in the load file, it is imported to also save the element structure. If IDs of a macro structure are changed, it is checked whether a file with subdivision data exists and asked to adjust the IDs.

Following dialog shows the available options:



Sort by coordinates

Nodes: Sort nodes

Elements: Sort elements

Order: Select order of axis, which the sorting scheme should follow. Alternatives are (xy), (yx), (xz), (zx), (yz), (zy), (xyz), (xzy), (yxz), (yzx), (zxy), (zyx). In case of giving only 2 axes there will be no sorting for the third direction. For example sorting scheme (xy) will be used for 3D-surfaces, which shouldn't be sorted in the z direction.

CoordSys: In case an ID of a local coordinate system is given, sorting is done within this system. The list box shows all defined coordinate systems.

Sort by

Group ID: Elements will be sorted in ascending order of their group Ids.

Mech.Type ID: Elements will be sorted in ascending order of their mechanical type ID.

Element ID: Elements will be sorted in ascending order of their ID.

Element selection

Only elements and pertinent nodes that are in the element selection will be considered.

Numbering:

ID1: Number for first element/node.

Inc: Increment for numbering.

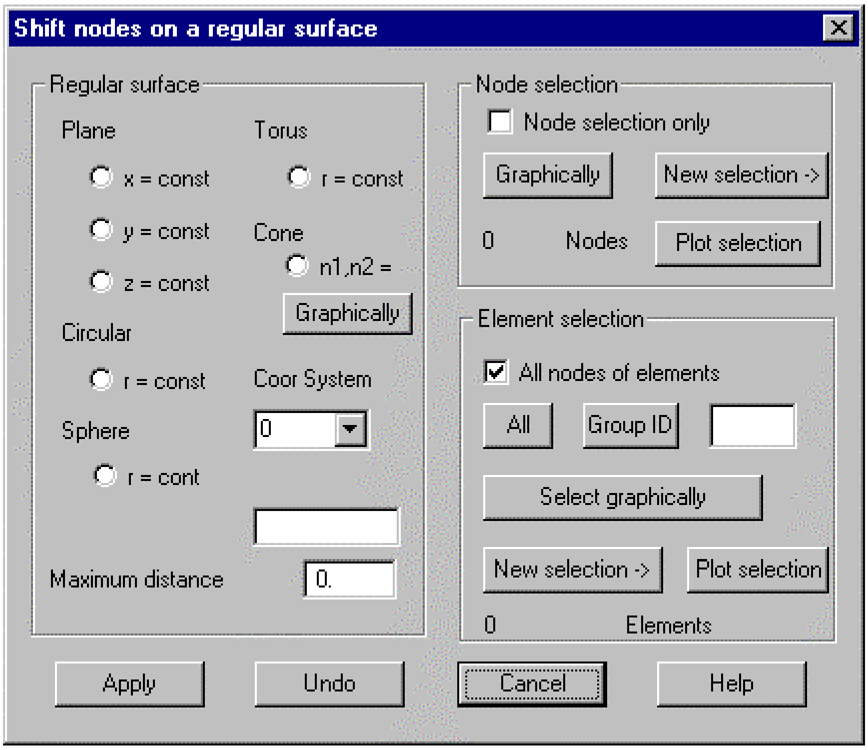
Smoothing: Shift nodes on a regular surface

With this command selected nodes can be moved to regular surfaces to smooth the structure. Smoothing especially will be necessary after subdivision of macro elements with curved surfaces where these curves are located on such regular surfaces. The calculation of additional internal nodes during this subdivision is based on C^0-Coons interpolation, which executes linear transition between opposite edges. Especially for nodes on spherical or toroidal surfaces the calculated nodes may more or less differ from the regular surface.

The regular surface must be defined within a local coordinate system. In case no local coordinate system is given the global system is used. After the definition of the surface all these nodes have to be selected which are to be moved onto this surface. With option "Maximum distance" a maximum distance may be specified. Nodes with distances from the surface greater than this value aren't moved. Moving is done perpendicular to the specified regular surface.

Moving of nodes can also be done by command **Intersection**, which does a translation of nodes along a vector and so an intersection of 2 regular surfaces can be achieved.

Following dialog shows the available options:



Regular surface

Plane: Any of x,y,z values can be set as a constant value in respect to a given coordinate system i.e. only the value in this direction is changed to the given value. The relevant value must be provided in the input field.

Circular cylinder: Radius must be given in the input field. Z-axis of the given coordinate system will be chosen as the axis of the cylinder.

Sphere: The radius of the sphere must be given in the input field. Basis of the given coordinate system will be chosen as the center of the sphere.

Torus: The radius of the toroidal surface must be given in the input field. Nodes P1 and P2 of the given coordinate system define the radius of the torus. This coordinate system must be defined as a toroidal system.

Cone: Two ID's of nodes lying on the cone surface must be given. If button "Graphically" is pressed, the 2 nodes can be selected graphically. The z-axis of the selected coordinate system will be used as the cone axis. The displacement of the nodes is perpendicular to the cone surface.

Coor System: The ID of a previously defined coordinate system must be given. The list box shows all currently available systems. ID 0 is for global coordinate system.

Input field: Parameters (for example radius, node ID's) must be given within this input field.

Maximum distance: Maximum value for the distance a node can be moved may be given.

Node selection / element selection

Select nodes to be translated or select elements where all nodes of selected elements are considered.

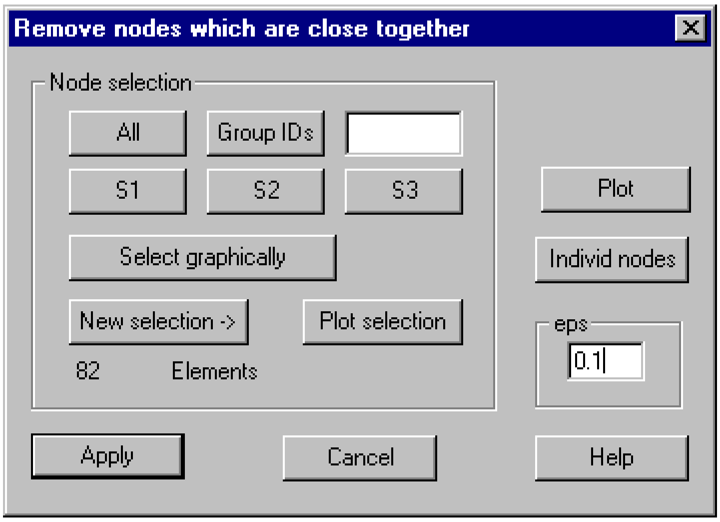
Undo

Pressing this button, the latest displacement is made undone.

Compress nodes: Remove nodes, which are close together

This command compresses the nodal data by eliminating nodes, which are close (within a given tolerance) together. Before compressing all related nodes can be checked graphically by pressing “Plot”. Pressing “Apply” causes the compression to be done.

Following dialog shows the available options:



Element selection

The nodes of all selected elements are considered.

Eps

The tolerance for nodes to be merged into one node must be given. Nodes with a distance lower than this value are merged into a single node. When reading a structure from AutoCAD compression is done automatically.

Plot

Nodes within given tolerance eps are marked by a colored symbol

Individual nodes

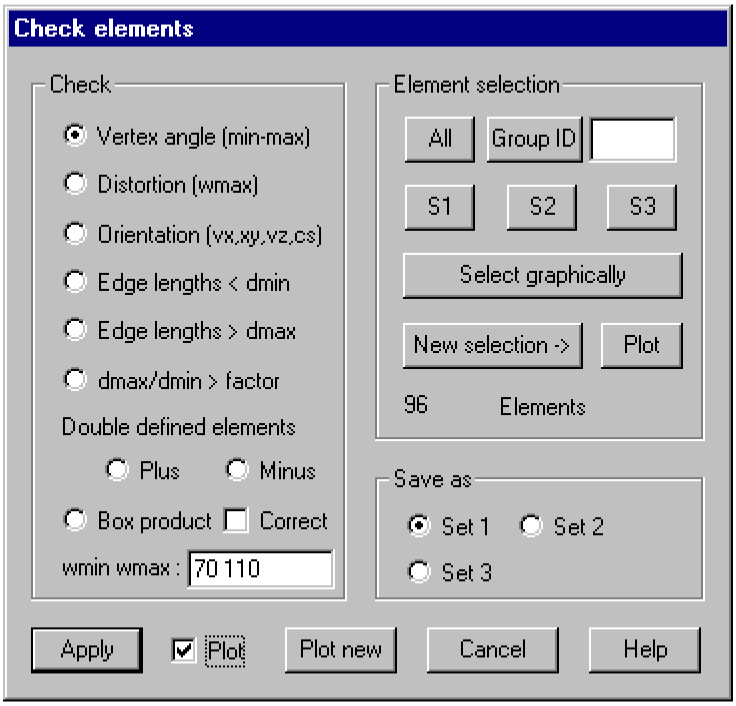
Nodes within given tolerance are marked. Afterwards two nodes at a time have to be selected graphically which are merged into a single node. Canceling is done by right mouse button.

Check elements: Check angles, orientation, etc.

Some checks on the finite element structure are done. The IDs of all the elements, which haven’t passed these tests, are saved as a new element selection set, these elements can also be graphically marked. This set can be modified later, for example by applying the command **Type2Type** for converting 4-node elements with very large or very small vertex angles into two 3-node elements.

The dialog remains active until pressing „Cancel“ closes it. After specifying the parameters the check is started using „Apply“.

Following dialog shows the available options:



Check

Vertex angle (min-max): Vertex angles of all selected elements are checked against the smallest and the largest permitted value, that have to be given in the input field.

Distortion (max): Distortion of 4-node elements is checked by calculating the angle between the normals in node 1 and 3 of the element. Maximum permitted value has to be given.

Orientation (vx,vy,vz,cs): The scalar product for the normal of the element and the given vector is calculated and checked for a positive sign. For all elements with negative results it's asked for turning the orientation in the opposite direction by reordering the vertices. Optional parameter cs can be the ID of a local coordinate system. In this case the orientation is calculated within this system.

Edge length < dmax: All elements are checked against an edge length lower than the given value.

Edge length > dmax: All elements are checked against an edge length greater than the given value.

dmax/dmin > fact: The ratio of the largest and the smallest edge length is calculated and checked against the given value.

Double defined elements: All elements are checked against each other whether there are elements with identical vertices. For this check it's distinguished between equal (Option "Plus") or opposite (Option "Minus") direction. In case double defined elements are found, the IDs of these elements are saved under Set1 and Set2. By selecting one of these sets afterwards the related elements can be removed with the command **Delete**. Please note that only surface elements are checked.

Box product: Box product (triple scalar product) for solid elements is checked at all corners for equal sign. The sign is taken from the first solid. If option „Correct" is marked the orientation of elements with a negative box product is automatically corrected.

Input field: Within this input field the numerical values for the above selected check must be given, angles are to be given in degrees.

Element selection

Only selected elements are checked.

Save as

Select the set (1 to 3) into which the IDs of the elements, which don't pass the checks, should be stored

Plot

With this option set all elements that do not pass the check are shown graphically by using a distinct color.

Plot new

This refreshes the graphics

Element Selection: Select elements to be affected by a command

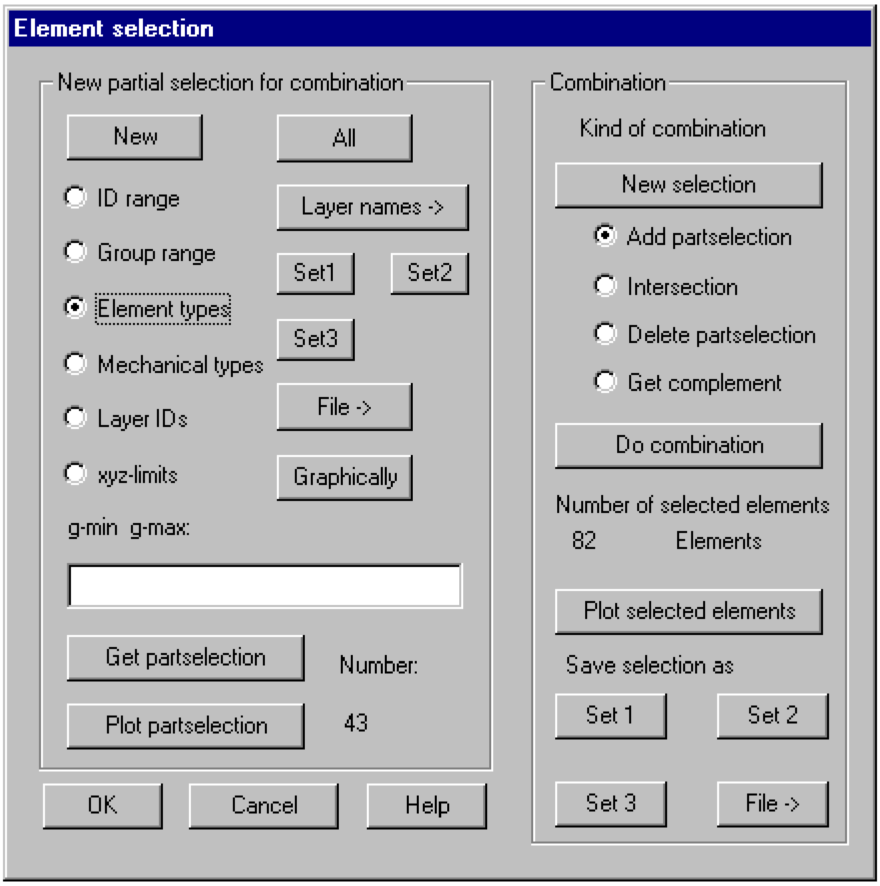
The dialog box will be popped up if inside another dialog the button “New selection” is pressed. This dialog also may be invoked independent of another command. In this case, it offers the opportunity of specifying and saving 3 different selection sets, which can be activated later by other commands. Element selection sets also may be stored permanently in a file.

The dialog remains active until pressing „OK“ or „Cancel“ explicitly close it. It’s possible to do multiple selections as partial selections which can be combined with the current global selection by multiple ways (see „Kind of combination“) The button “get part selection” carries out such a partial selection according to the given parameters and shows the number of elements included. The dialog always shows the current number of selected elements.

By pressing “Plot” the current partial selection respectively the current global selection can be checked graphically. Within the graphical window the center of gravity of all selected elements is marked.

Button „Do combination“ combines the current partial selection with the currently saved global selection using the marked option and produces a new global selection. The number of selected elements is shown. Default for the global selection is always the latest selection except the latest selection contained all elements. In this case the global selection is set to zero.

Following options are available:



New partial selection for combination

For the specifications of partial selections the following alternatives are available:

New: Erase actual partial selection.

All: All macro respectively finite elements currently in memory are selected.

ID range: First and last ID of a range of elements must be given.

Group range: First and last group ID of a range of groups must be given.

Element types: Lowest and highest geometrical type ID must be given. All elements with type ID within this range are selected.

Mechanical types: Lowest and highest mechanical type ID must be given. All elements with type ID within this range are selected.

Layer ID: The lowest and the highest layer ID must be given. All elements with assigned layer ID’s within this range are selected.

xyz-limits: These elements with all nodes inside the given coordinates are selected. The range must be given with xl,xr,yl,yr,zl,zr,cs where l,r stands for left and right coordinate limits and cs is the (optional) ID of a coordinate system. With a given coordinate system all nodes are transformed to this system prior to checking.

Layer names: Within a list box all defined layer names are shown. Multiple names can be selected.

Set1 - Set3: Selection sets previously defined and saved (see “Save as”) are used.

File: Load selection sets from file. The file name has to be selected from a dialog.

Graphically: Following selection has to be done graphically within the OpenGL window. All selectable elements are marked by a symbol in the center of gravity. A dialog box pops up where you should choose the kind of the graphically selection.

Input field: Within this input field you have to specify the parameters needed for some of the above options. The values are separated by space or comma.

Get part selection: After specifying the options for determining partial selections this button causes the calculation of the partial selection. The number of found elements is shown within the dialog.

Plot part selection: Pressing this button will mark all elements contained in the current part selection with a symbol.

Kind of combination

The current partial selection can be combined with the current global selection. The following alternatives are available for this combination:

New selection: The currently stored global selection is deleted.

Add part selection: Merge partial selection with global selection.

Intersection: Only elements within both selections build up the new global selection.

Delete part selection: Elements, which are in the partial selection, are removed from the global selection.

Get complement: Only complementary elements, that is only these elements not contained within the partial selection build up the global selection.

Do combination: Pressing this button will cause the combination of the partial and global selection to be done.

Plot selected elements: Pressing this button will mark all elements contained in the current global selection with a symbol.

Save selection as

Set 1- Set 3: The current global selection will be saved into the chosen set and can later be used by other commands. Because external element ID's are saved these sets are only valid until element numbering won't be changed by deleting or sorting the elements.

File: Currently in sets 1 - 3 stored selections are saved in a file. The file name has to be given in a file selection dialog, it gets the extension .ews.

Node Selection: Select nodes to be affected by a command

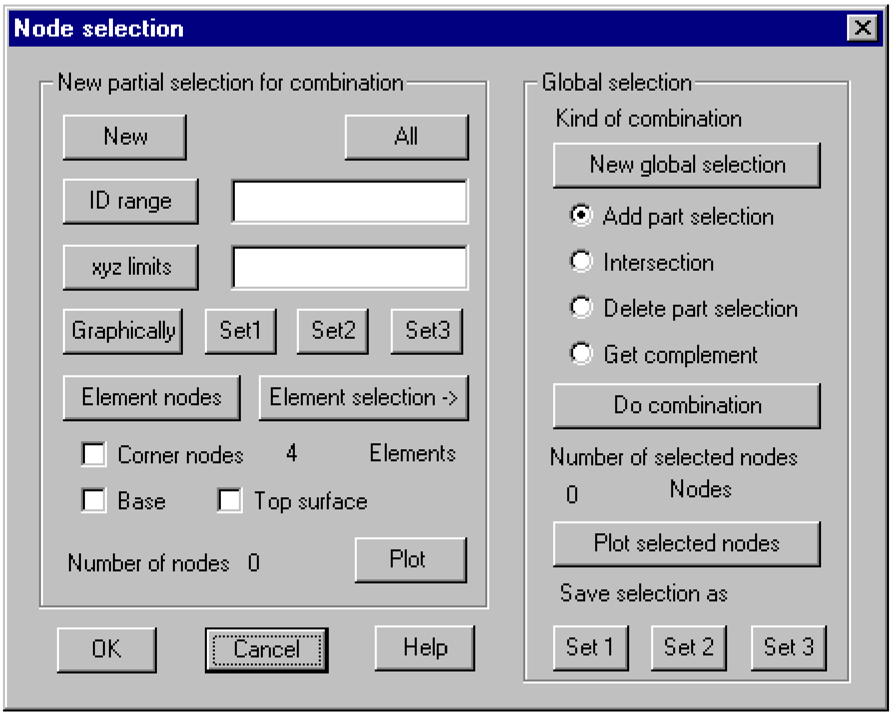
The dialog box will be popped up if inside another dialog the button “New selection” is pressed for getting a node selection. This dialog may also be invoked independent of another command. It offers the possibility of specifying and saving 3 different selection sets, which can later be activated by other commands.

The dialog remains active until it's explicitly closed by "OK" or "Cancel". New partial selections can be specified multiple times, which can be combined with the current global selection by different ways (see "Kind of combination").

By pressing the “Plot” button the current selection can be graphically checked. Within a new display list selected nodes are marked with a colored symbol.

Button „Do combination“ combines the current partial selection with the global selection using the given parameters and produces a new global selection. The number of selected nodes is shown. Default for the global selection is always the latest chosen selection except the latest selection contained all nodes. In this case the global selection is set to zero.

Following options are available:



New partial selection for combination

For specifying a new partial selection for a combination with the global selection the following alternatives are available:

New: Erase current partial selection.

All: All nodes are selected.

ID range: First and last ID of a range of IDs must be given.

xyz limits: All nodes are selected which are inside a given cube. Input parameters for the cube are xl, xr, yl, yr, zl, zr, cs that are the left and right coordinates of the cube within a given coordinate system cs.

Set1-Set3: Actual saved selection within this set is used.

Graphically: Following selection has to be done graphically within the OpenGL window. All selectable nodes are marked by a symbol. A dialog box pops up where you should choose the kind of the graphically selection.

Element nodes: All nodes of the current selected elements are selected. With option „Corner nodes“ marked only element vertices are selected without nodes on edges, with option „Base“ or „Top surface“ marked, only node of the base or top surface of solid elements are selected, With button „Element selection“ the active element selection used by this command can prior be defined.

Plot: Pressing this button plots the partial selection.

Kind of combination

Following alternatives for the combination of a new partial selection with the global selection are available:

New global selection: The current global selection will be erased.

Add pat selection: New partially selected nodes are merged with the global selection.

Intersection: The intersection between new partial selection and global selection builds the new global selection, i.e. only those nodes contained within both sets are selected.

Delete part selection: New partial selected nodes are removed from the current selection.

Get complement: A complement set is build, i.e. all currently not selected nodes are contained within the new global selection.

Do combination: Pressing this button will cause the combination of the partial and global selection to be done.

Plot selected nodes: Pressing this button, the current global selection can be checked graphically.

Save selection as

The current selection will be saved within the chosen set and can be used by following commands. Saved are the external node ID's, i.e. the set remains valid until numbering won't be changed.

MAKROS: Give Command selection to MAKROS menu

After starting MAKROS from within AutoCAD the menu bar of AutoCAD is extended by the commands MAKROS and xunload. With this command program control switches to MAKROS where a menu bar is shown with MAKROS commands. Afterwards its no longer possible to execute commands from within AutoCAD respectively entered commands won't be processed until control is given back to AutoCAD with the command "AutoCAD".

AutoCAD: Command selection from AutoCAD menus

After switching command selection to MAKROS with command **MAKROS,** program control can be switched back to **AutoCAD** by giving this command.

xunload: Exit MAKROS

Using the AutoCAD command (xunload "makrosae") exits MAKROS if it was started from within AutoCAD with (xload "makrosae"). The ARX version of MAKROS cannot be unloaded; it stays active until AutoCAD is finished. If the structure has been changed it will be asked for saving which will eventually invoke the **Save** command

Commands for graphical representation of the structure

Overview

Menu group „**Graphic**" contains the following commands:

Plot structure	Plot elements
Plot IDs	Plot group IDs etc.
View point	Define view parameters
Rotation	Define view by rotating around model axis
Layer	Erase AutoCAD layer respectively display-lists or switch display-lists on or off
Light source	Set shading parameters
Color definition	Define colors
Post color	Define color values for color indices
Graphic text	Define additional graphical text
Select font	Define font for graphical text
Postscript	Write graphics to a postscript file

Command description

Plot: Plot elements

The command plots selected elements within one or distinct AutoCAD layers respectively display lists. It's possible to distinguish between different kinds of plots ranging from wire frame to surface plots. In all cases curved edges are approximated by polylines where additional nodes between vertices are automatically calculated to smooth the plot. This can be suppressed when transferring elements to AutoCAD. It should be suppressed when these elements are intended to be read back (after a modification within AutoCAD) by MAKROS.

Option „Edges only" draws the edges of the elements as wire frames. Surface elements will be drawn as closed polylines. When transferred to AutoCAD, solids of type 8x are drawn with 9 edges within one polyline, the remaining 3 vertical edges are drawn as separate polylines. All these elements can be re-read with the command **Read AutoCAD.**

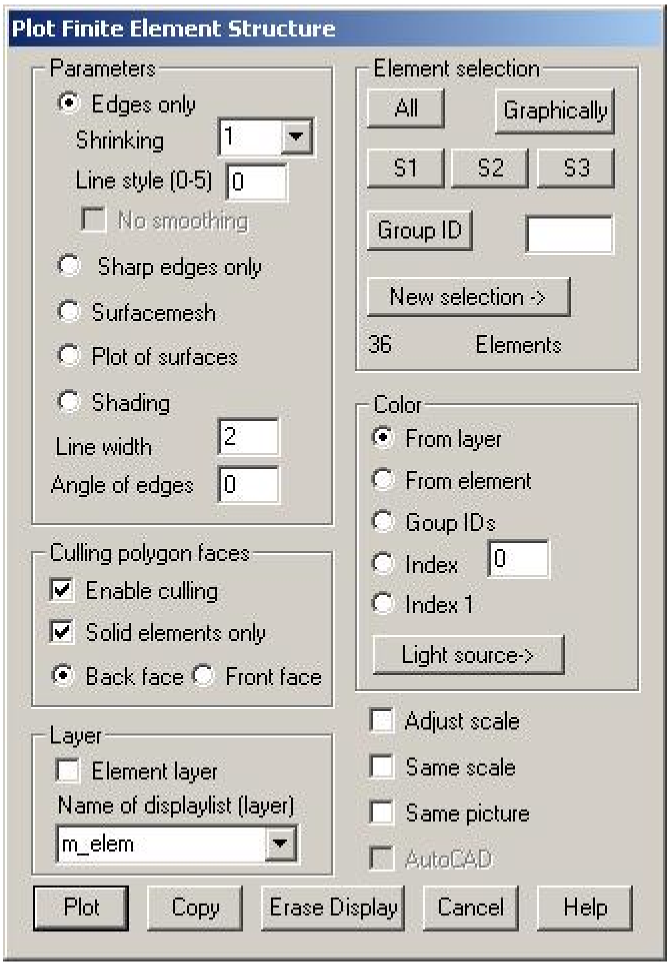
Option „Surface mesh“ plots all selected elements as an AutoCAD mesh. This option can only be used during a graphical plot within AutoCAD. It is only implemented for elements of type 30 and 40. This kind of plotting has the advantage that all elements are corrected when moving common nodes. This isn't the case in edge plot mode where all relevant elements are drawn as separate polylines. Surface meshes can be read again as a macro or FE model. In case of a large structure it should be divided by suitable element selections into several plots.

Option „Sharp edges only“ plots edges where the angle between the normal vectors of the adjacent surfaces is larger than the given value. Only these edges and the edges with only one adjacent surface are plotted. This way it's easy to detect gaps within the structure. Solid structures are represented only by their outer surfaces. The angle must be provided in degrees.

Option „Plot of surfaces“ plots all elements in AutoCAD as a polygon mesh. Solid structures are represented only by their outer surfaces. With this kind of plot AutoCAD rendering can be used for hidden line removal or shading. Elements plotted this way cannot be read back with the command **Read AutoCAD.** Graphical outputs within an OpenGL window automatically suppresses hidden lines and surfaces, curved surfaces are approximated by triangles. It’s also possible to shade the structure using a specified light source.

Pressing button “Plot” starts the graphical output. By pressing button „Erase Display“ the graphics window will be erased (including all OpenGL display lists). Button „Copy“ causes a new window to be opened with a copy of the existing window so it will be possible to compare multiple views of the structure.

Following Dialog shows the available options:



Parameters

<u>*Edges only:*</u> Edges of the elements are plotted as polylines. With „No smoothing” set, no additional nodes on curved edges will be generated when transferring the elements to AutoCAD.

<u>*Shrinking:*</u> Factors < 1.0 reduces elements in size when plotted by shrinking them towards there center.

Line style: When plotting edges a line style (0-5) used for all edges can be set.

Sharp edges only: Only sharp edges of the structure are plotted. The max. angle between normal vectors must be given.

Surface mesh: Elements are plotted as AutoCAD 3D mesh (only for types 30 and 40 and option "AutoCAD").

Plot of surfaces: The outer surface of the element structure is approximated by triangles and hidden line removal is done. An angle of sharp edges can also be given.

Shading: One or more light sources (see **Light source**) may be witched on to produce a shaded image.

Line width: In the input field, the width of the lines in pixels can be given.

Angle on edges: If an angle > 0 is given when using surface or shaded plots only those edges between surfaces with an angle between their normal vectors greater than this value will be plotted. This value must also be set when using "sharp edge plots".

Culling polygon faces

Enable culling: If this option is marked culling facility of OpenGL is switched on, so that only faces with normal vector directed to the camera position (option back faces) or opposite (option front faces) are plotted.

Solid elements only: If this option is also marked, culling is only applied to faces that belong to solid elements.

Layer

Element layer: With this option set, all elements will be drawn in different layers respectively display lists. There must have been made an association of layer numbers and layer names to the elements. OpenGL graphics use one display list for each layer.

New layer: With this option set, new names for layers will be created and used when transferring the structure to AutoCAD.

Using new layer names is importing, when transferring elements to AutoCAD, because existing entities of the used layers will at first be erased. With OpenGL, using different display lists makes it possible to turn the plot of parts of the structure on and off.

Element selection

Only elements contained in the selection are plotted.

Color

From layer: The color indices associated with the layers will be used.

From element: Previously assigned element color is used.

Group IDs: The elements group ID will be used as a color index.

Index: All elements are plotted with this index.

Light source: Pressing this button, a dialog pops up where a light source can be specified.

Adjust scale

With this option set scaling and center of the image will be automatically adapted for the next plot based on the size of the current graphics window. Otherwise new scaling is only performed when changing the current element selection.

Same scale

Each time a new element selection is drawn, the reference point and scale factor are newly calculated. With this option set the old view is used instead. So different parts of the structure can be superimposed with different colors on different display lists.

Same picture

By default all display lists respectively used AutoCAD layers are erased before plotting. With this option set erasing won't be done. When using OpenGL another display list must be specified, a previously used display list can't be extended.

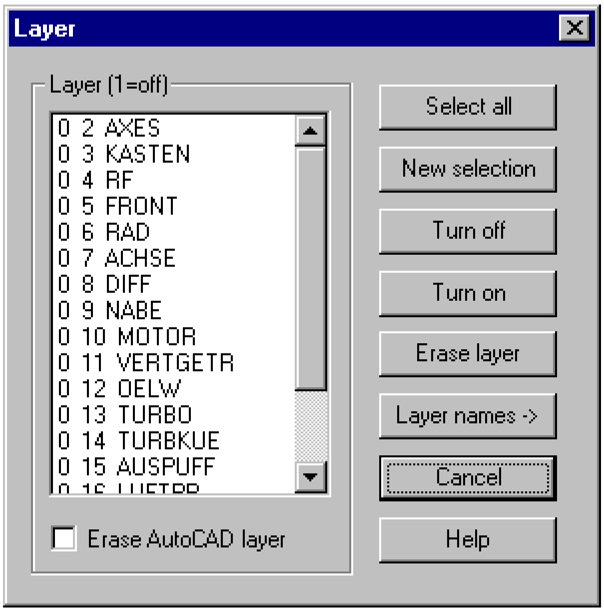
AutoCAD

With this option active selected elements are transferred to and plotted within AutoCAD. Otherwise the OpenGL graphics window is used. This option is only available after starting MAKROS from within AutoCAD.

Layer: Erase layers or turn OpenGL display lists on or off

With this command you can erase all entities on given AutoCAD layers. It's also possible to turn OpenGL display lists on and off.

Following Dialog shows the available options:



Layer

The list shows all currently plotted display lists respectively existing layers within AutoCAD. Digit 0 or 1 within the first column shows the status (on or off) of the associated display list.

Erase AutoCAD layer

With this option set, the list only shows layers currently defined within AutoCAD. It's possible to select these layers and to remove entities on these layers with button „Erase layer".

Select all

All layers within the list are selected.

New selection

The current selection is canceled.

Turn off

Turn off selected display lists.

Turn on

Turn on selected display lists.

Erase layer

The selected display lists respectfully AutoCAD layers are erased. Entities on AutoCAD layers are deleted.

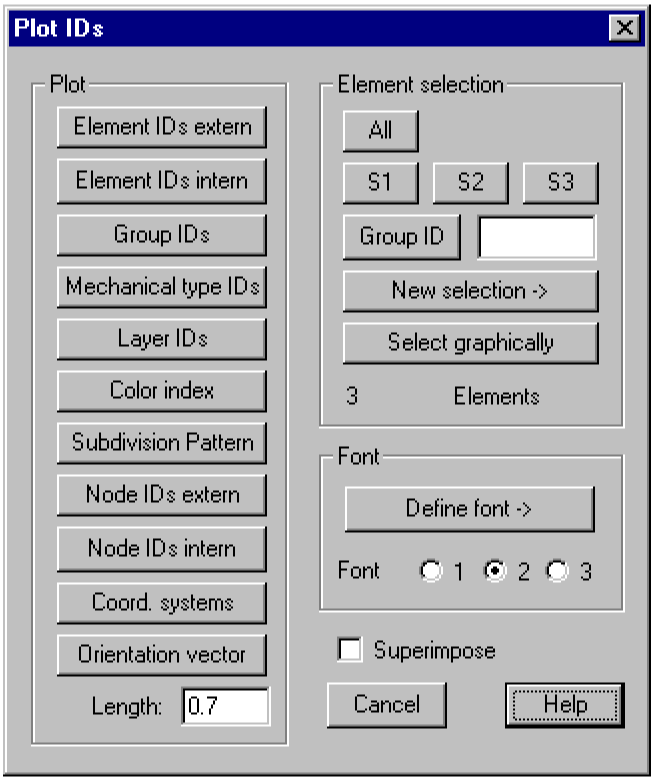
Layer names

A dialog pops up to define new layer names.

Plot ID: Plot IDs (type, group, etc.)

With this command assignment of IDs (element, node, group, etc.) can be checked graphically. The IDs are shown within an additional layer.

Following dialog shows the available options:



Plot:

Element IDs extern / intern: The external respectively internal IDs of the elements will be plotted.

Group ID: The assigned group IDs will be plotted.

Mechanical type ID: The assigned mechanical type IDs will be plotted.

Layer ID: The assigned layer IDs will be plotted.

Color index: The assigned color indices will be plotted.

Subdivision pattern: The assigned subdivision patterns (for macro elements only) will be plotted.

Node Ids extern / intern: The external respectively internal IDs of the nodes will be plotted.

Coord. systems: The axis of currently available coordinate systems and the IDs will be plotted.

Orientation vector: In the center of gravity of surface elements an orientation vector will be plotted which shows the orientation of this element. The length of the vectors must be given in the input field.

Element selection

Select elements for witch the IDs should be plotted.

Font

The font, which should be used to display text, can be selected.

By pressing the button „Define font“ a dialog pops up where a font and text color can be assigned to the font IDs 1-3.

Camera position: Define camera position

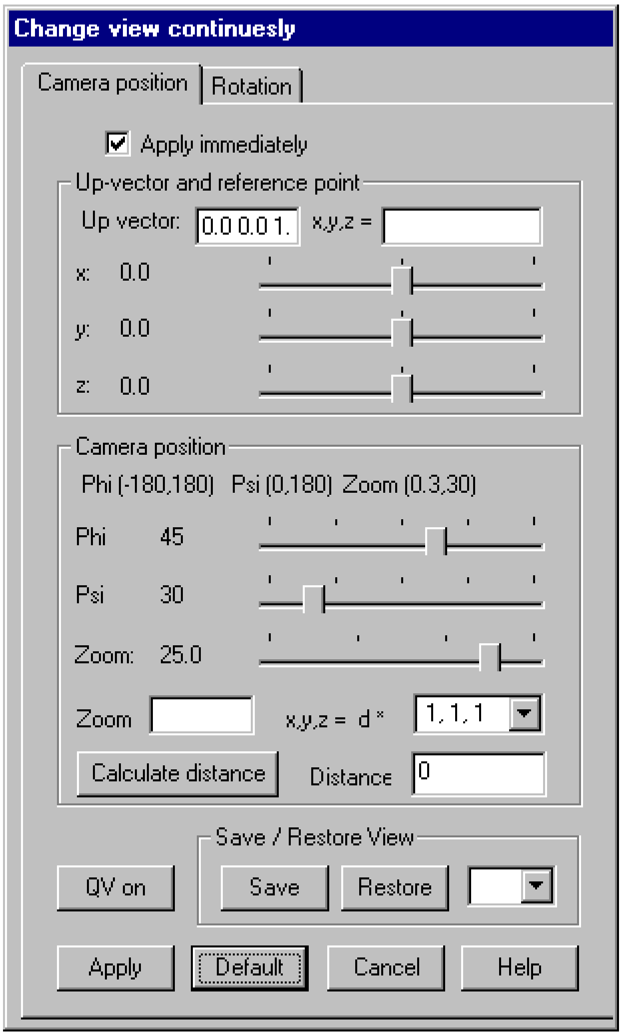
The dialog defines a camera position within polar coordinates, a target point within model coordinates and a model orientation given by an up-vector. These parameters can be dynamically changed by sliders (using mouse or cursor keys).

Hint: dragging the pointer in the graphics window with the left button pressed can also dynamically change Angles Phi and Psi. Dragging horizontally will change angle Phi, dragging vertically will change angle Psi.

When altering the view the display lists aren’t freshly build, only the view is adopted. So a quick display can be reached. When moving the sliders, the view is immediately adjusted only if the option “Apply immediately” is checked. With complex graphics each altering of the view may require several seconds, in this case adjusting the view may be quickened with button “QV on” (see below):

The zoom angle and the distance of the viewpoint may also be given numerically. From the list also 8 camera position in the 8 quadrants of the coordinate system can be selected. To adjust the view the button “Apply” must be pressed. Button “Default” sets a default view in the first quadrant.

Following Dialog shows the available options:



Apply immediately

With this option set the graphical representation will be immediately actualized after a slider is moved. Disabling this option several parameters can be modified before pressing button „Apply“ generates a new plot.

Up-vector and reference point

Reference point (target point) is specified with x,y,z-coordinates in model space. Default is the center of the shown model. The sliders can move this point. The up-vector defines the axis of the coordinate system, which should be drawn upwards. Default is (0,0,1), i.e. z-axis upwards. In case the camera is positioned along the z-axis (Psi = 0 or 180 degrees), default up-vector will be (0,1,0) (y-axis upwards).

Up vector and reference point can also be given numerically in the input fields, updating is done after pressing button “Apply”.

Camera position

Phi: Angle of rotation within x-y-plane between -180^{o} and $+180^{o.}$

Psi: Angle of rotation measured from z-axis in the range of 0^{o} to $180^{o.}$

Zoom: Angle of aperture in the range of 0.3^{o} (large) to 30^{o} (small). As numerical value can also be given in the input field.

Calculate distance: Pressing this button, a default distance between camera and target point is newly calculated. A new value can also be given in the input field.

_X,y,z = d *:_ The list shows some predefined points for the camera position in the global coordinate system. After selecting one option the view is adjusted by pressing button “Apply”.

Default:

Pressing this button, the default view is newly applied.

QV on / of

This button allows a quicker finding of the optional view (QV stands for Quick View) in case the graphics is very complex as with shading or representation of scalar fields. The button functions as a switch and is alternately labeled „QV on“ and „QV off“. Switching QV on causes all currently visible display lists to be set invisible and a new list to be displayed that shows only the sharp edges of the selected elements. This reduced graphics can quickly be rotated or zoomed to find the desired view. After that QV is to be switched of with the same button and the full graphics will be visible again.

Save / Restore view

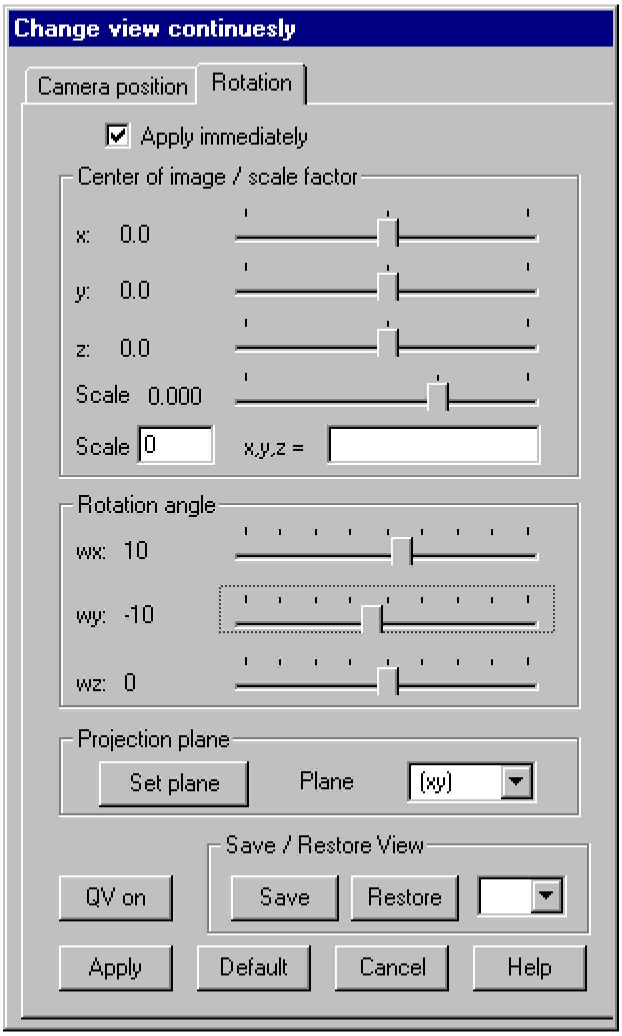
Pressing button “Save” the parameters of the current view will be stored, pressing button „Restore“ can later restore this view. Up to 4 different views can be stored; the number of view must be given in the input field.

Rotation: Define view by rotating around model axis

Within this dialog the projection plane, center of plot, scale factor and rotation of the model around global axis can be specified. The sliders can dynamically modify these settings.

Hint: dragging the pointer in the graphics window with the left button pressed can also dynamically change Angles wx and wy. Dragging horizontally will change angle wy, dragging vertically will change angle wx.

Following Dialog shows the available options:



Apply immediately

With this option set the graphical representation will be immediately actualized after a slider is moved. Disabling this option, several parameters can be modified before pressing „Apply“ generates a new plot.

Center of image / scale factor

Global x, y, z-coordinates for the center of the plot and the scale factor must be given. Default will be the center of the current selected elements with a scaling so that all selected elements fits within the window. By reducing the scale factor or moving the center of the plot, parts of the model can be zoomed. With the sliders center and scale can be changed continuously.

Center of image and scale factor can also be given numerically in the input fields, updating are done after pressing button “Apply”.

Rotation angle

These sliders define rotations about the global coordinate axis in the range of -90^{o} to $+90^{o}$

Projection plane

Out of the drop down list you can select one of the following projection planes: (xy), (xz), (yz), -(xy), -(xz), -(yz). With projection planes having a negative sign, the view is in the direction of the positive coordinate axis perpendicular to the projection plane, otherwise it is in the opposite direction. Pressing „Set plane“ does updating.

Default

The default view is applied.

QV on / of

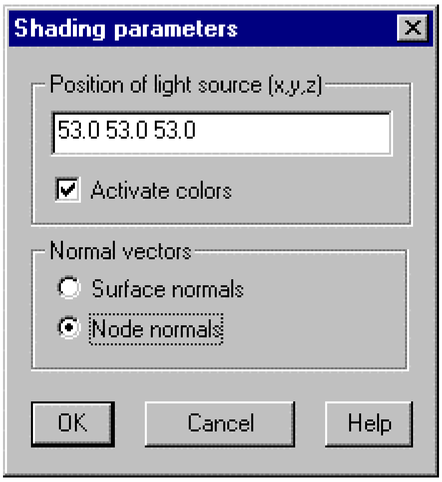
This button allows a quicker finding of the optional view (QV stands for Quick View) in case the graphics is very complex as with shading or representation of scalar fields. The button functions as a switch and is alternately labeled „QV on“ and „QV off“. Switching QV on causes all currently visible display lists to be set invisible and a new list to be displayed that shows only the sharp edges of the selected elements. This reduced graphics can quickly be rotated or zoomed to find the desired view. After that QV is to be switched of with the same button and the full graphics will be visible again.

Save / Restore view

Pressing button “Save” the parameters of the current view will be stored, pressing button „Restore“ can later restore this view. Up to 4 different views can be stored; the number of view must be given in the input field.

Light source: Define shading parameters

For displaying shaded structures, the position of the light source and the kind of normal vectors to be used must be set in the following dialog:



Position of light source (x, y, z)

Give the position of the light source in the input field.

Activate colors: With this option set the elements are plotted with associated colors. Otherwise grayscales are used.

Normal vectors

For calculation of light intensities normal vectors of the surfaces are needed. Therefore it’s important that all surfaces use correct node ordering (see command **Check**

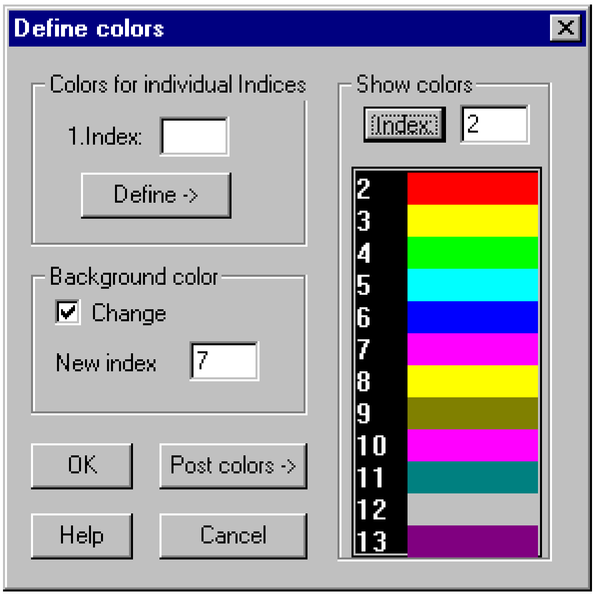
Surface normals: Only the normal vector of element surface is used for calculation of light intensity.

Node normals: On each node a normal vector is calculated as a mean value of all adjacent elements, this gives a smooth transition of intensities among elements.

Color Definition: Define colors

This command can define colors for graphics within the OpenGL window or change the background color. Color for indices 0 and 1 cannot be changed. Index 0 is used for current background (Default: black) and index 1 is current foreground (Default: white).

Following Dialog shows the available options:



Colors for individual indices

1. Index: Within this input field the first color index to be changed must be provided. Changeable are only indices starting at 2.

„Define“ pops up the standard Windows dialog for defining colors. 16 colors starting from the specified first index are shown with their current values, which can be changed.

Background color

Change: This option sets the background color to the color of the specified index. Index 0 means black and index 1 means white background.

Show colors

Within the input field a color index must be given. The corresponding list shows the currently defined colors starting from this index, after pressing the associated button.

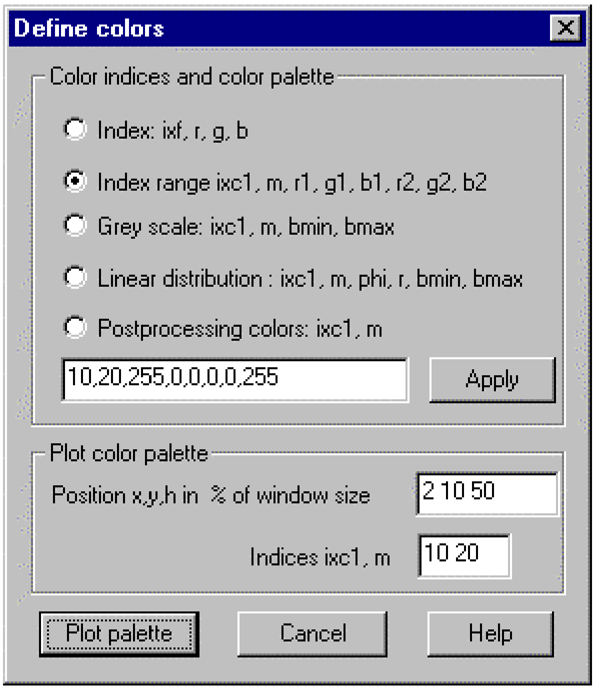
Post colors

Pressing this button a dialog pops up for defining smooth color transitions used by post processing.

Post color: Define color values for color indices

This command redefines colors associated to color indices.

Following Dialog shows the available options:



Index: ixc, r, g, b

Color index and associated RGB-values in the range 0-255 must be given for a single index

Index range: ixc1, m, r1, g1, b1, r2, g2, b2

The smallest color index ixc1 and the number of indices for which new values should be defined together with RGB-values for the smallest and the largest color index must be given. Color values for all indices between are calculated using linear interpolation.

Grey scale: ixc1, m, bmin, bmax

The smallest color index ixc1 and the number of color indices m must be given together with the brightness values in % for the smallest and the largest index. Indices between are calculated using linear interpolation.

Linear: ixc1, m, phi, s, bmin, bmax

The smallest color index ixc1, the number of indices m, the color value phi, the color saturation s, the smallest brightness bmin and the largest brightness bmax must be given. Color is defined by an angle phi (0-360), a radius r (0-100) out of a color cone (HLS-system).

Following colors are assigned to angles: 0° = blue, 60° = magenta, 120° = red, 180° = yellow, 240° = green, 300° = cyan, 360° = blue.

Post processing colors: ixc1, m

The smallest color index and the number of following indices must be given. Between these indices a smooth color range of colors blue, green, yellow, red and magenta will be created.

Apply

After setting the options this button does the current calculation of the colors and updates the color lookup tables.

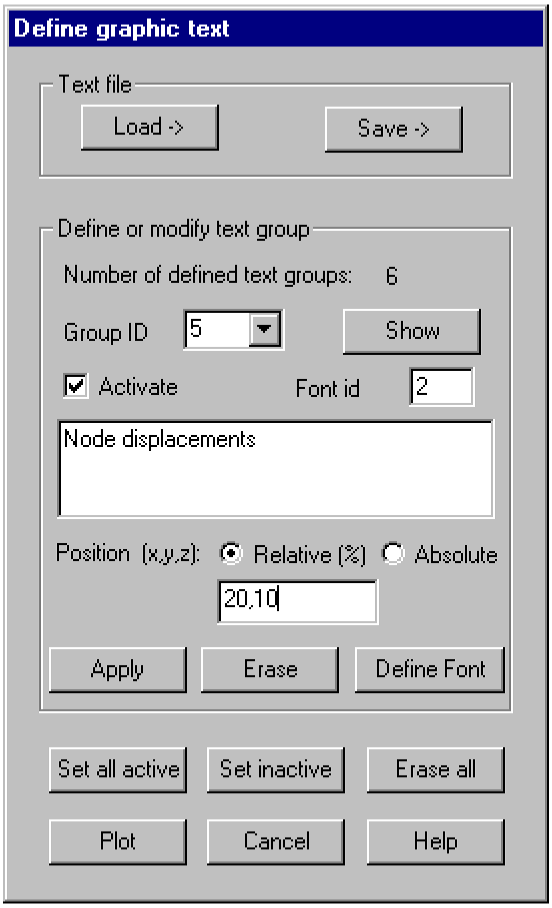
Plot palette

This button plots current color settings as a color bar for indices ixc1 until ixc1+m-1. The position (x,y) and hight h of the color bar must also be given in % of window size.

Graphical text: Define additional text to be displayed in the graphics window.

Text can be added to graphics. Graphical text can be defined within several text groups which can be activated or deactivated and can be saved in files with .txt extension.

Following Dialog shows the available options:



Text file

Load: Loads a text file. If there are already text groups available the loaded groups are appended.

Save: Saves all defined text groups to a file.

Define or modify a text group

Group ID: a unique ID identifies Text groups. Within the list box all currently defined IDs are shown.

Show: Currently defined parameters for the selected text group are shown and can be modified.

Activate: With this option set the text group will be plotted during graphics, otherwise it is not plotted.

Font ID: Selected ID of the font to be used when plotting this text group (1-3).

Text input field: Up to 3 lines of text can be defined for the selected text group. Pressing Ctrl + Return inserts a line break.

Position: The position of the text group can optionally be provided within relative coordinates (relative to the size of the graphics window in %) or as absolute coordinates within model space.

Apply: After setting the parameters for a text group the group must be explicitly saved by pressing button „Apply".

Erase: The selected text group will be completely deleted.

Define font: Font and color to be used by a font ID can be selected out of a dialog box.

Set all active

All text groups are set active.

Set all inactive

All text groups are set inactive.

Erase all

All currently defined text groups are deleted.

Plot

All active text groups are plotted within a separate display list which isn't erased when the graphics window is erased so it may be necessary explicitly to set text groups inactive.

Select Font: Define font for graphical text

In a Dialog color and font height for three different font Ids can be chosen.

Postscript: Write graphics to a postscript file

OpenGL graphics does not support graphics output to a printer. This command allows creating a postscript file. Multiple images (up to 6) may be arranged on one single page, the number must be given; the images are automatically positioned and have equal size if the number is even.

After specifying the file title and some parameters, the postscript file is opened and the postscript header is written by pressing the button „Open file". Optionally some own postscript commands can be copied from a given file on top of the postscript file to be created. After opening, the graphics window is hidden and graphics into the file is done using the MAKROS graphics commands as usual. Graphic is superimposed until a new image is selected with button "New image", For example structure plot, plot of node Ids and plot of graphic text are done with different commands. For graphic text the text groups that should be plotted must be set active before. Text height and color is taken from the chosen font.

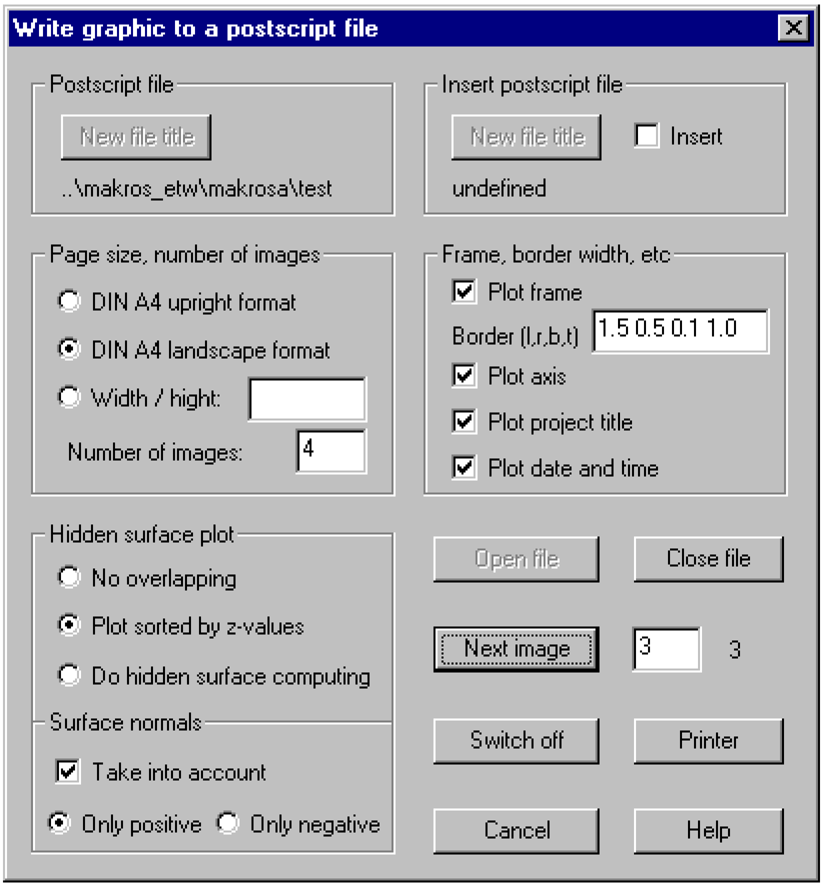
Button „Close file" must be used to close the postscript file; with button „Printer" it can immediately be sent to a printer. If more than one image are arranged on a page, the active image must be finished and the next image opened by using button „Next image".

For postscript the window coordinates are scaled, if some lines or surfaces are only partly visible in the graphics window these lines and surfaces are not plotted at all, because their coordinates are not available.

If the used printer does not support color, grayscale should be used for surface plots. Grayscales can be defined using command „Post color". If a black background is used for graphics in the graphics window grayscales are automatically adjusted for postscript. Color index 0 is used for white and color index 1 is used for black.

With some kinds of graphics representation it may be better to make a hardcopy from the graphics window using HyperSnap and print this copy, for example column plot douse no hidden line calculation.

After invoking the command, the shown dialog box pops up with following options:



Postscript file

The actual title of the postscript file is shown in the dialog, default is projectname.ps. Using button “New file title” a new file can be selected in a file selection dialog. The selected file will get the extension .ps.

Insert Postscript file

If the option “Insert” is set, the postscript commands contained in the give file will be copied on the top of the file to be created. Using button “New file title”, a new file can be selected in a file selection dialog.

Page size, number of images

DIN A4: Page size is set to DIN A4 upright format.

DIN A4 landscape format: Page size is set to DIN A4 90° rotated; all images are also rotated by 90°.

Width / height: Width and height of page size can be given in cm in the input field.

Number of images: The number of images to be positioned on one page can be given in the input field (≤ 6).

Frame, border width, etc.

Plot frame: If this option is set, each image will be framed.

Border width: In the input field, the width of the borders on the left, right, bottom and top of the page can be given.

Plot axis: With this option set axis are plotted to show the current view.

Plot project title: If this option is set, the project title is plotted on the left of the top border.

Plot date and time: If this option is set, the date and time of file creation are plotted on the right of the top border.

Hidden surface plot

For the plot of surfaces it is important to know how the surfaces are overlapping each other. It can be chosen between following options:

No overlapping: This option is fastest, the surfaces are continuously plotted.

Plot sorted by z-values: The surfaces are plotted corresponding to their z-position, surfaces with lower z-positions at first.

Do hidden surface calculations: This option is most time consuming, a hidden surface calculation is done by software. The used algorithm assumes that there are no intersections between surfaces. Pixel coordinates of the actual graphic are used for hidden surface calculation. To avoid rounding errors the graphic window should be as big as possible if the structure has lots of small elements.

Surface normals

The plot of surfaces will be less time consuming, if some surfaces can be separated by the direction of the normal vectors. For example this is the case, if a closed or a solid structure is plotted where all surfaces on the backside can be separated.

Following options can be selected:

Take into account: The normal vectors of the surfaces is checked

Only positive: All surfaces with normals oriented to the back are separated.

Only negative: All surfaces with normals oriented to the front are separated.

Open file

Pressing this button, a new postscript file with the given title is opened and a new page is initialized. After that, graphics can be written to this file.

Next image

If more than one images are positioned on one page, this button has to be used to initialize the next image, the number of this image has to be given in the input field. The number of the active image is shown in the dialog.

Close file

Pressing this button the postscript file is closed and can be sent to the printer.

Switch off / on

This button can be used to interrupt the output to the postscript file, for example to change and test the view for the next image. When interrupting output to the file, the graphics window is set visible and following graphics will be shown in the graphics window. With the same button, which is now labeled “Switch on” graphics output is newly directed to the postscript file and the graphics window is hidden again.

Printer

Using this button, a newly created postscript file can immediately be sent to the printer. If the file is still opened it will be closed. For printing, there must exist a file “maka_interface.ini” in the folder of the executable, with the following line:

Printer: “printer command”

For “printer command” the text enclosed in quotation marks must be given to send a file to the printer, where a % character is the placeholder for the file title, for example

Printer: “lpr printername %”

Printer: “copy % COM2”

The given text will then be executed as a system command. The file “maka_interface.ini” is also used for special interfaces (see chapter “Interfaces to FE-Programs“)

Commands specifying loads, constraints, element properties and materials

Overview

For the definition of loads, constraints, element properties and materials, commands are provided that allow the definition of the most important data types used by the “Neutral PATRAN Interface” and the “NASTRAN Interface” (see chapter “Interfaces to FE-Programs“).

The following commands are available:

Node forces	Define and assign forces or displacements on nodes
Element loads	Define and assign distributed loads on elements
Distributed loads	Define distributed loads on surface elements, calculate statically equivalent forces
Surface pressure	Define constant pressure on surface elements, calculate statically equivalent forces
Resultant	Calculate the resultant of node forces
Material	Define material properties
Cross-section	Define and assign cross section properties on elements
Constraints	Define and assign constraints on nodes
Check NASTRAN	Check assignments for the NASTRAN interface
Save to file	Save data to a binary file
Load from file	Load data from a binary file
Plot load data	Plot assignments
Record	Show some information of assignments
Delete all	Delete all data in memory

Command “**Node forces**” serves to assign forces or moments or displacements to individual nodes. With command “**Element loads**” individual elements are assigned distributed loads corresponding to the NASTRAN interface. Command “**Distributed loads**” allows defining a distributed load on element surfaces that is immediately transformed to statically equivalent node forces on the corner nodes of the elements. Command “**Surface pressure**” allows assigning elements a constant pressure normal to the element surface where the elements may be three dimensionally curved; this pressure is also immediately transformed to statically equivalent node forces on the corner nodes of the elements.

The data defined with these commands is stored in a binary file with extension .lqd.

These commands are only available if a FE-structure is loaded and processed. Node forces, element loads and properties are assigned to nodes and elements of the currently processed FE-structure where the external IDs of the nodes and elements are used, therefore these external IDs should not be changed after loads and constraints are defined.

Node forces and constraints may be defined in reference to a local coordinate system where the vector components are interpreted as

$$\mathbf{v} = v_1\mathbf{e}_1 + v_2\mathbf{e}_2 + v_3\mathbf{e}_3$$

With v_1, v_2, v_3 = given vector components and $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$ = basis vectors of the global or a local coordinate system. The basis vectors are node dependent with cylindrical and spherical coordinate systems and oriented in direction of increasing coordinate values.

If node forces or moments are written into a NASTRAN- or PATRAN-file it can be specified that the vectors defined in local coordinate systems should at first be transferred into the global coordinate system.

Command description

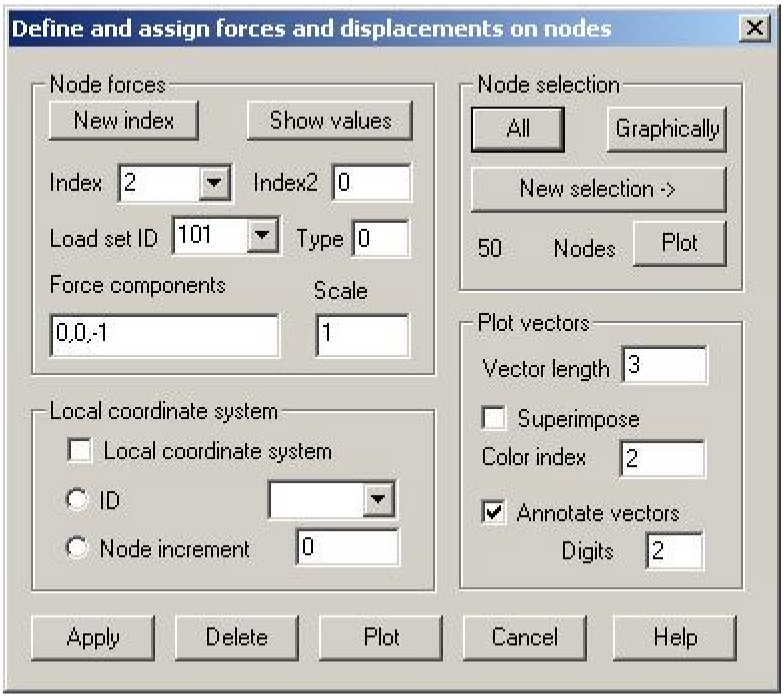
Node forces: Define and assign forces and displacements on nodes

Each force or displacement is defined by 3 vector components. A single force can be assigned to multiple nodes. Parameters of the force definition and the assignments to nodes are saved within a single load group. Each group will be assigned 2 IDs to identify it: one index and one load set ID. The load set ID can be identical for several groups; it will be saved within the PATRAN and NASTRAN interface and can be used by the FE program for example as an ID for a load case. Each group however has its own unique index. This index isn't saved by the PATRAN or NASTRAN interface; it only serves as an identification of load groups when defining loads. All values saved under such an index can be deleted at all and newly redefined. After saving such a group the index for the next group is automatically increased by 1. If an index of an already defined group is entered within the input field, all the corresponding values of this group are shown within the relevant fields of the dialog by pressing button "Show values", so these values can easily be modified, plotted or removed. The button „New index" causes a new index to be created.

The vector components may be specified in reference to a local coordinate system (for example radial loads within a cylindrical coordinate system). This coordinate system has to be defined before; the ID of the system has to be given. When the PATRAN or NASTRAN file is created all vectors defined within a local coordinate system may optionally be transferred to the global system.

The button „Apply" saves the specified forces definition. In case there are already values stored with this index it will be asked for overwriting. Answering „No" will cancel this operation and the old values are kept. By pressing button „Delete" saved values with the shown index are deleted. Pressing „Plot" does a plot of the vectors.

After invoking this command the following dialog is popped up with the shown options available. The dialog remains active until it's explicitly closed with button „Cancel".



Node forces

Index: The index of an already defined or a new load group must be given. A given number greater than the largest used number + 1 automatically reduces the number to this value.

New index: Next available index is set in the input field.

Index2: For the buttons „Delete“ and „Plot“ an upper limit of a range of indices can be given in this input field.

Show values: Pressing this button, the stored values of the group with the given index are displayed in the different input fields.

Load set ID: A load set ID must be given. Within the list box all previously used IDs are shown. It’s possible to assign an already defined load set ID to a new load group.

Type: With a type ID, different types of vectors can be distinguished. For the NASTRAN interface the following type IDs are to be used: 0 = node force, 1 = node moment, 2 = node displacement, 3 = node rotation.

Force components: Up to 3 values can be given. Lacking values are set to 0.

Scale: For the NASTRAN interface a scale factor can be given in the input field.

Local coordinate system

Local coordinate system: The option has to be set, if vector components are given in reference to a local coordinate system.

ID: In case of a definition of the vector components within a local coordinate system the ID of this system must be given here. Within the list box all available IDs are shown and can be selected. ID 0 is used for the global coordinate system.

Node increment: In case the loads are defined within local coordinate systems, which were created by the option „Boundary curve“ within command Coordinate System, the increment must be given which was used when defining the coordinate systems.

Node selection

These nodes must be selected to which the given vector should be assigned. The IDs of these nodes are stored with the load group.

Plot vectors

Vector length: The length for displaying the largest saved vector must be given.

Superimpose: With this option set a new display list will be used for plotting, so it’s possible to plot the vectors of different groups together by using different colors.

Color index: With different color indices different load groups can be distinguished graphically.

Plot

The button invokes a plot of the vectors of all groups within the given range of indices.

Delete

Pressing the button deletes all stored data for the given range of indices.

Element load: Define and assign distributed loads on elements

This command allows the definition of constant or distributed element loads corresponding to the PATRAN interface. For each element load up to 6 components may be given. Which components are given defines a component flag (parameter ICOMP in PATRAN interface), for example ICOMP = 13 (stands for the PATRAN flags 101000) means that the load consists of a force component for the x- and z-direction.

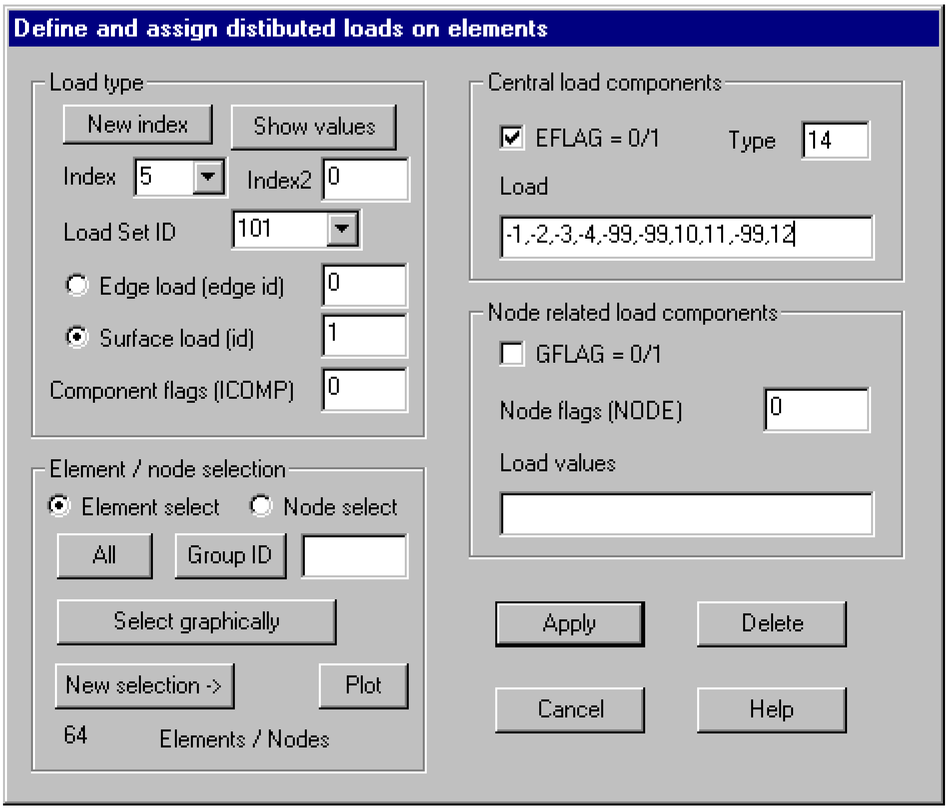
For the PATRAN interface it’s possible to define central or node related element loads. For central loads NC load values must be specified, where NC means the number of active flags within ICOMP. For node related loads node flags (parameter NODE in PATRAN interface) specifies to which element nodes load values were assigned, for example NODE = 1234 (stands for the PATRAN flags 11110000) means that load values are specified for the 4 corners of an element of type 42. The number of values to be provided is NN*NC where NC is the number of activated flags in ICOMP and NN the number of activated flags in NODE. Order is NC values for each node.

For the NASTRAN interface only a load set ID, a type ID and load values in the input field for central load components are to be given. With a type ID different types of loads are distinguished (see chapter “Interfaces to FE-Programs“). All other given parameters in the dialog (ICOMP, NODE, etc) are not used.

Each group of element loads is assigned a pair of identifiers: one index and a load set ID. The load set ID can be identical for different groups but each group must have its own index. When an existing index is given within the corresponding input field all associated values of that group can be displayed by pressing button “Show values”. These values can then be modified, plotted or even deleted. Button „New index“ generates a new index for the next load group.

Button „Apply“ saves specified values. Button „Delete“ deletes previously saved values for a given index or range of indices.

Following Dialog shows the available options, the dialog remains active until pressing button „Cancel“ ends it.



Load type

Index: The index of an already defined or a new load group must be given. A given number greater than the largest used number + 1 automatically reduces the number to this value.

New index: Next available index is set in the input field.

Index2: For the buttons „Delete“ and „Plot“ an upper limit of a range of indices can be given in this input field.

Show values: Pressing this button, the stored values for the given index are displayed in the dialog.

Load Set ID: An ID for the load set must be given. The list box shows all previously used IDs.

Edge load: For edge loads specify the ID of the associated element edge.

Surface loads: For surface loads on solid elements specify the surface ID (1-12) of the solid element for which this load should be applied.

Component flags (ICOMP): In a number containing the digits 1-6 must be specified, for which directions load components are given (for example 135 results to the flags 101010).

Central load components

EFLAG: This flag has to be set if central load components should be saved.

Type: With a type ID different types of element loads may be distinguished (see NASTRAN interface).

Load values: A load value has to be given for each direction that is turned on in the flag ICOMP. For NASTRAN, the number of load values depends on the given type ID; the flags are not used (see NASTRAN interface).

Node related load components:

GFLAG: This flag has to be set if node related load components should be saved (not used with NASTRAN).

Node flags (NODE): In a number containing the digits 1-8 must be specified, for which element nodes load values are given (for example 1234 results to the flags 11110000).

Load values: For each activated node in the flag NODE NC load values must be given, where NC = number of activated components within ICOMP.

Element / node selection:

Alternatively an element selection or a node selection may be specified. If a node selection is given, all elements are searched for whose corner nodes are contained in the given node selection. In case of solid elements, the surface of the solid is searched whose corner nodes are contained in the node selection, the ID of this surface is used for the interface.

Surface loads: Define distributed loads on surface elements, calculate statically equivalent forces

The command allows do define constant or variably distributed loads on a plane or a cylindrical surface whereby the distributed load is immediately converted to statically equivalent forces on the corner nodes of the loaded elements.

The area and the direction of the distributed load are defined in a local coordinate system that is determined by 3 element nodes. The line from the first to the second node gives the local x-axis. The given 3 nodes also define the maximal expansion of the load area; it must completely be contained within the limits of the local coordinates given by the 3 points.

At first, all elements or surfaces of solid elements are determined, that lie on the plane and within the expansion of the local coordinate system, only the corner nodes of these elements respectively element surfaces are considered.

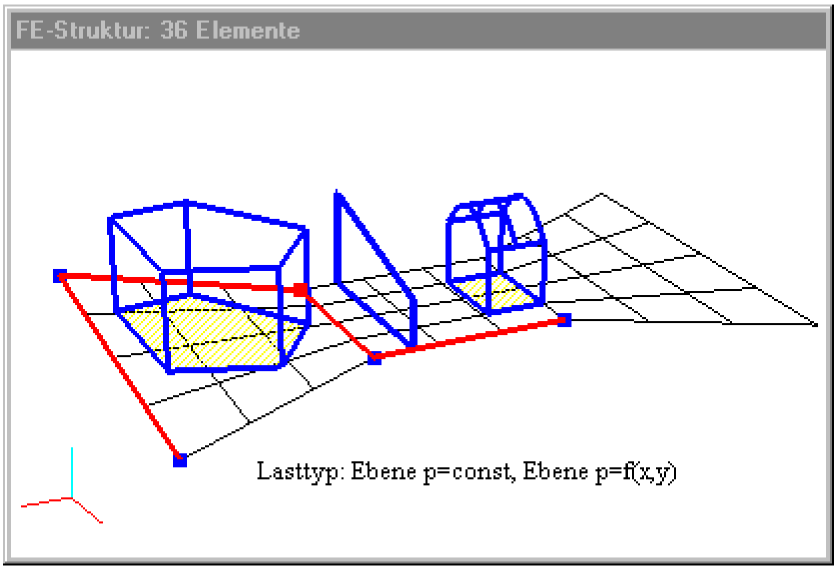
Four types of surface loads are distinguished:

Plane, p = constant

With this option, the loaded area may be a convex polygon of up to 8 points. Only constant load distribution is allowed (see following picture).

Plane, p = f (x, y)

With this option, the loaded area must be rectangular and parallel to the axis of the local coordinate system. All loaded elements must also be rectangular and parallel to the axis of the local coordinate system. The distributed load may be variable in both directions (see following picture).

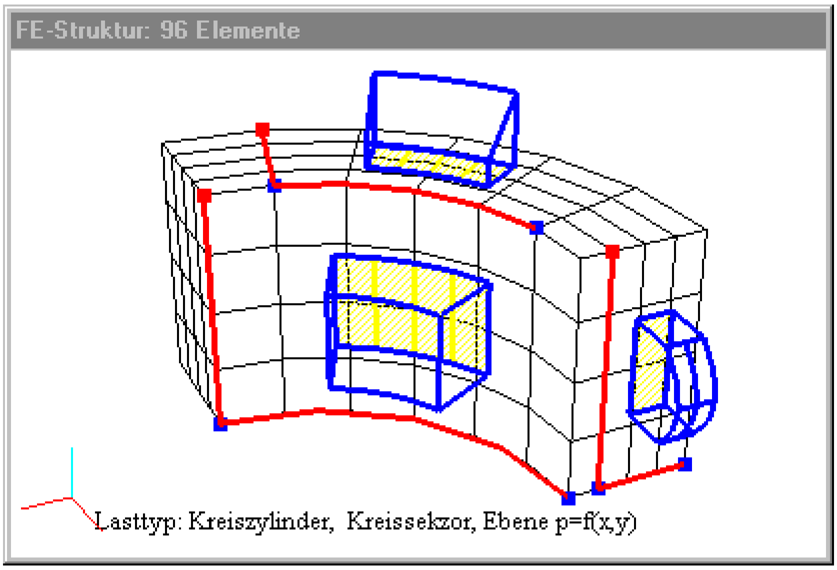


Section of a circle

With this option, a rectangular section of a circle is loaded. Additional to the 3 reference nodes the ID of a defined cylindrical coordinate system must be given. All edges of loaded elements must be parallel to the axis of the given cylindrical system. The distributed load may be variable in both directions (see following picture).

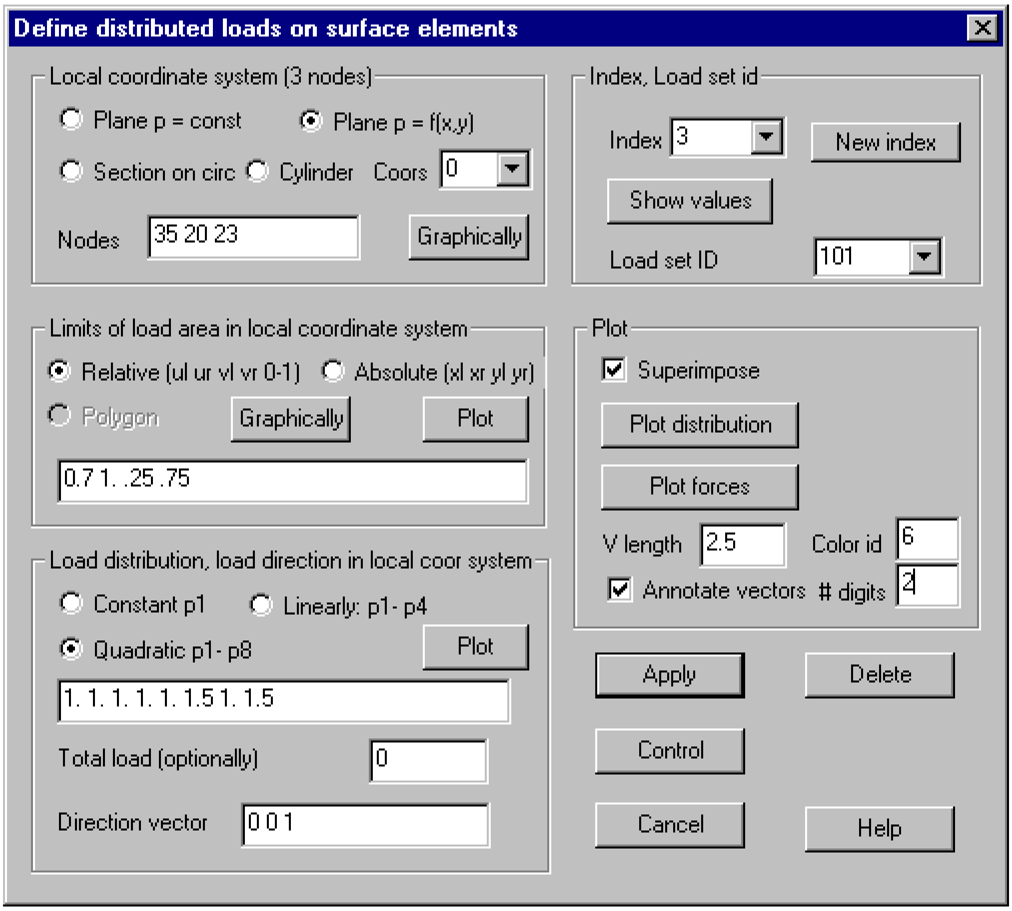
Cylindrical surface

With this option, a rectangular section of a cylindrical surface is loaded. Additional to the 3 reference nodes the ID of a defined cylindrical coordinate system must be given. All edges of loaded elements must be parallel to the axis of the given cylindrical system. The distributed load may be variable in both directions (see following picture).



Each defined surface load is identified by 2 IDs, an index and a load set ID. The load set ID may be the same for different surface loads, but each surface load has a unique index. Pressing button “New index” gets the next unused index. Loads defined with command **Element loads** and **Surface pressure** are stored in the same data structure and use the same continuous range of indices. If the index of a stored surface load is given in the corresponding input field, with button “Show values” the stored values are displayed in the dialog box. These values may be plotted, deleted or modified. With button “Apply” the given values in the dialog are stored and the statically equivalent node forces are calculated and stored in the data structure. With button “Delete” stored values may be deleted.

Following dialog shows the available options:



Local coordinate system (3 nodes)

<u>Plane (p = constant)</u>: A rectangular or polygonal bounded area with a constant load can be defined.

<u>Plane (p = f (x, y))</u>: A rectangular area with a variably distributed load can be defined. Edges of the load area and of the elements must be parallel to the axis of the local coordinate system.

<u>Section of a circle</u>: A rectangular area of a circle with a variably distributed load can be defined. Additionally to the 3 nodes, the ID of a defined cylindrical coordinate system must be given; the list box shows all defined coordinate systems. Edges of the loaded area and of the elements must be parallel to the axis of the cylindrical coordinate system.

<u>Cylindrical surface</u>: A rectangular area of a cylindrical surface with variably distributed loads can be defined. Additionally to the 3 nodes, the ID of a defined cylindrical coordinate system must be given; the list box shows all defined coordinate systems. Edges of the loaded area and of the elements must be parallel to the axis of the cylindrical coordinate system.

<u>Nodes</u>: The IDs of 3 nodes to define the local coordinate system must be given. After pressing button “Graphically” the 3 nodes can be selected graphically.

Limits of load area in the local coordinate system.

<u>Relative (ul, ur, vl, vr)</u>: The left and right limits of the rectangular load area must be given relative to the length of the axis of the local coordinate system ($0 \le u, v \le 1$). The left and right limits may be the same to define a line or point load.

<u>Absolute (xl, xr, yl, yr)</u>: The left and right limits of the rectangular load area must be given in coordinates of the local coordinate system.

<u>Polygon</u>: The (u, v) coordinates of up to 8 vertices of a polygon must be given relative to the length of the axis of the local coordinate system ($0 \le u, v \le 1$). The load area must be convex. This option is only available for a local coordinate system of type “Plane, p = constant”.

Graphically: After pressing this button, left and right limits of a rectangular area respectively of up to 8 vertices of a polygonal area can be selected graphically. The relative coordinates of the selected points are shown in the input field.

Plot: Pressing this button, the expansion of the loaded area is shown graphically.

Load distribution, load direction in the local coordinate system.

Constant: The value of a constant load distribution must be given.

Linearly: The load values at the 4 corners in the counter clockwise order must be given.

Quadratic: Additionally to the 4 values by linearly distribution 4 load values at the means of the 4 edges of the rectangular load area must be given.

Plot: Pressing this button, the load distribution is shown graphically.

Total load: If a value > 0 is given in the input field, the calculated forces are scaled so that the amount of the resultant is equal to this value.

Direction vector: The direction of the calculated force vectors must be given relative to the local coordinate system.

Index, Load set ID.

Index: The index of a new or already defined load group must be given. The list box shows all previously used indices for this kind of distributed loads. For the buttons “Plot” and “Delete” 2 indices of a range of indices may be given.

New index: Pressing this button the next unused index is shown in the input field for a new load group.

Show values: If the index of an already defined load group is given, the stored values of this group are displayed in the dialog after pressing this button.

Load set ID: A load set ID must be given for a new load group. The list box shows all previously used load set IDs.

Plot

Superimpose: If this option is marked, a new display list is used for plotting.

Plot load distribution: For the given index or range of indices the following is plotted: reference nodes and axis of a local coordinate system, load area and load distribution.

Plot forces: For the given index or range of indices the calculated statically equivalent forces are plotted.

Vlength: The plotted length of the largest vector respectively the maximal height for the load distribution must be given.

Color index: The color index for the plot must be given.

Annotate vectors: If this option is set, the vectors are labeled with the amount of the force vector.

Digits: If the vectors are to be labeled, the number of digits after the decimal point must be given.

Control

Pressing this button, the resultant of the force vectors is calculated. The components of the resultant and its position relative to the local coordinate system are displayed in the protocol window. The resultant is also graphically displayed.

Apply

The given values are stored in the internal data structure; the statically equivalent node forces are also calculated and stored. If there are already values stored under the given index, it is asked for overwriting.

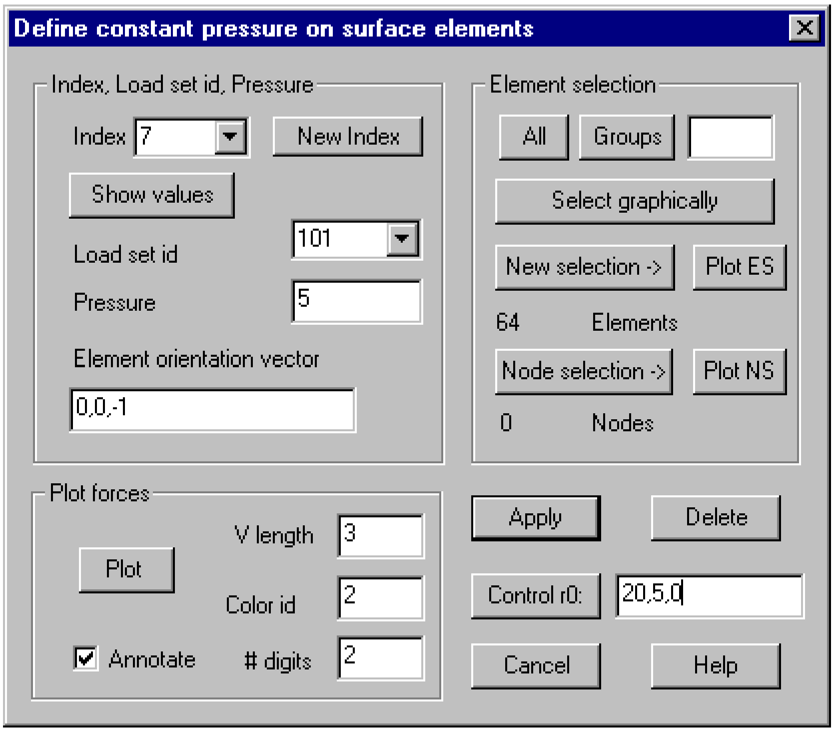
Delete

All stored values for the given index or range of indices are deleted.

Surface pressure: Define constant pressure on surface elements, calculate statically equivalent forces

With this command a constant pressure on the surface of elements may be defined. The pressure is immediately converted to statically equivalent forces on the corner nodes of the elements. The elements can be three dimensionally curved, the direction of the pressure is always perpendicular to the surface points, and the integration is done numerically.

The following dialog shows the available options:



Index, Load set ID, Pressure

Index: The index of a new or already defined load group must be given. The list box shows all for this kind of load previously used indices.

New index: Pressing this button, the next free index is displayed in the input field.

Show values: If the index of an already defined load group is given, pressing this button will display all stored values for the index in the dialog.

Load set ID: A load set ID must be given, the list box shows all previously used indices.

Pressure: The amount of pressure must be given.

Element orientation vector: Positive pressure is in direction of the normal of the element, because of that the order of the element nodes is important. If a global vector is given in the input field, the orientation of the element definition is checked. In case the scalar product of the given vector and the normal of the element is negative, the orientation of the element is turned around while calculating the node forces.

Element selection

The elements must be selected that should be loaded with the pressure. In case of solid elements additionally to the element selection a node selection must be given. Only the surfaces with all corner nodes contained in the node selection are used. Buttons “PlotES” and “PlotNS” allows graphically checking the element respectively node selection.

Plot forces

Pressing this button, the given values are stored in the internal data structure; the statically equivalent node forces are also calculated and stored. In case there are already values stored under the given index, it is asked for overwriting.

Delete

All stored values for the given index or range of indices are deleted.

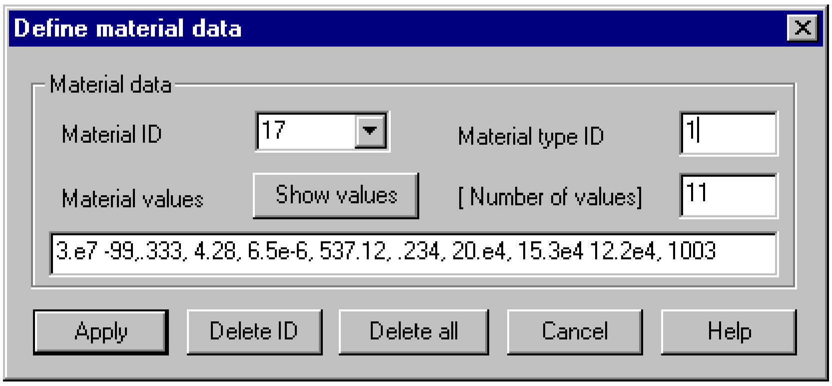
Control r0

Pressing this button, the resultant of all node forces and the resultant moment of the forces in reference to the origin of the global coordinate system or in reference to a given point (r0) are calculated and displayed in the protocol window. In the input field the global coordinates of a reference point for the resultant moment may be given.

Material : Define material properties

This command defines material properties. The meaning of these values follows PATRAN neutral file format respectively is described in the NASTRAN interface for different types of materials. A material ID and a material type ID identify different materials. In case a material ID is given for which there are already values stored these values are shown within the related input fields when the button “Show values” is pressed. Button „Apply“ saves these values and „Delete“ deletes already saved values. Button “Delete all” deletes all stored material data.

Following Dialog shows the available options:



Material data

Material ID: A unique material ID has to be given. The list box shows the previously used IDs.

Material type ID: A type ID in the range 1 to 13 when using PATRAN interface respectively 1 to 18 when using NASTRAN interface must be given.

Number of values: If a number is given here it is check whether the number of given material values in the next input field is correct.

Material values: PATRAN interface saves up to 96 material values. Lacking values are set to 0. The number of values that are needed for different NASTRAN materials is described in chapter “NASTRAN – Interface”.

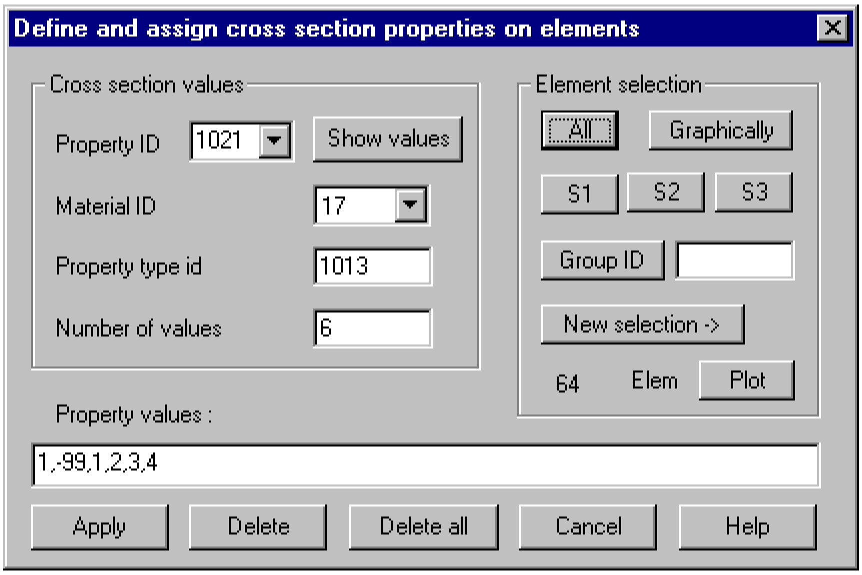
Cross sections: Define and associate cross section or other properties

This command defines the cross section and other properties and associates them to elements. The properties are saved as „Element properties“ within PATRAN interface. For the NASTRAN interface the command is used to define properties and to define additional element parameters that are needed for some NASTRAN element types. Each cross section is assigned a unique ID. If the ID of an already saved cross-section is given, button “Show values” displays the corresponding values in the dialog. So once saved properties can be easily reused and assigned to a new element selection or modified.

With command **Plot** or **Check NASTRAN** it can be checked whether all elements have been assigned a proper cross-section.

The dialog remains active until it’s explicitly closed by „Cancel“. Button „Apply“ saves current values and „Delete“ deletes already saved values. „Delete all“ deletes all stored cross sections.

Following Dialog shows the available options:



Cross section values

Property ID: The unique ID for this property must be given.

Material ID: The ID for the associated material set has to be given. This ID was previously set by the command **Material**. The IDs of defined materials are shown within the list box.

Property type ID: The given number is saved as parameter N3 when using the PATRAN interface. Using the NASTRAN interface, the property type must be identical to the mechanical type ID of the elements + 1000 when defining values for the NASTRAN property record and it must be equal to the mechanical type ID of the elements when defining additional parameters of the elements (see NASTRAN interface):

Number of values: The number of associated cross section values must be given.

Property values: The values for this cross section must be given in the input field.

Element selection

The elements have to be selected, to which this cross section should be associated.

Constraints: Define and assign constraints on nodes and additional node data

With this command constraints and additional node data are defined, that are saved within PATRAN interface as „Node Data“. For all nodes with no associations made the following defaults apply (within parentheses the related PATRAN variable is shown)

Node type (GTYP): G (structural nodes)

Number degrees of freedom (NDF): 6

Node class (CONFIG): 0

Coordinate system (CID): 0

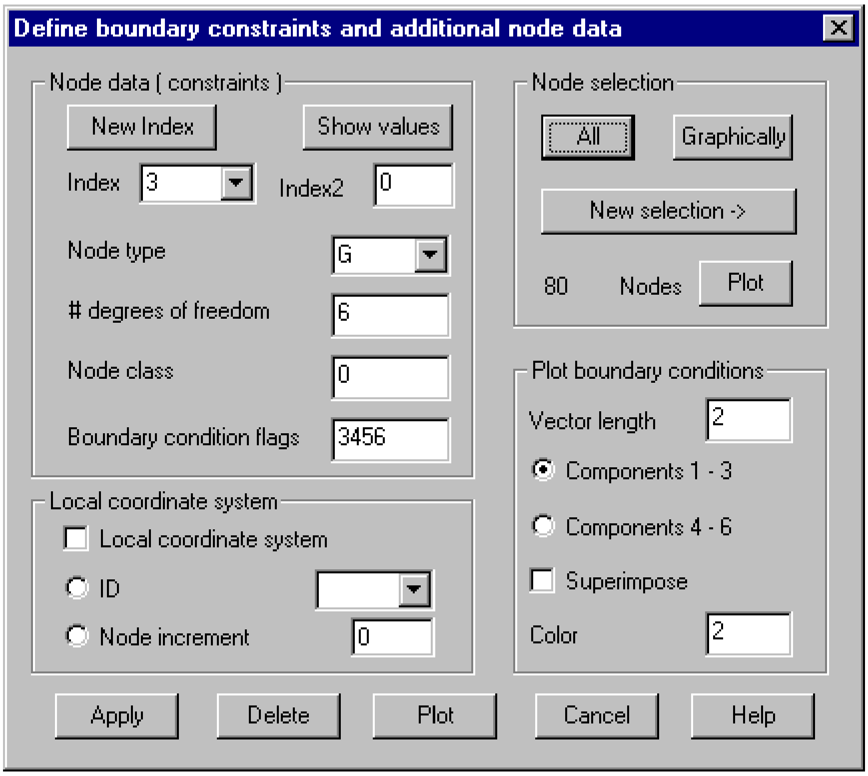
Constraint flags (PSPC): 000000

With the NASTRAN interface, only the boundary condition flags and the local coordinate system are used, these values are stored optionally in the GRID records or in SPC1 records.

Different definitions are identified by a sequenced index. The values defined under such an index can be completely deleted and new defined. After saving a definition the index is increased by one. After entering an already defined index all associated values are shown within the dialog after pressing button "Show values" and can be deleted or plotted. Button „New index“ gives the next unique index. The constraints can be defined within a local coordinate system (for example radial components within a cylindrical coordinate system). The coordinate system must have been already defined.

Button „Apply“ saves the specified values for this definition. If there are already values stored under this index it will be asked for overwriting. Button „Delete“ deletes already saved values. Button „Plot“ plots boundary condition. The dialog remains active until it's explicitly closed by „Cancel“.

Following Dialog shows the available options:



Node data (constraints)

<u>*Index*</u>: The index of a new or of an already saved definition must be given.

<u>*New index*</u>: Next available index is set.

<u>*Show values*</u>: The values of an already saved definition are displayed in the dialog.

<u>*Index2*</u>: The upper limit of a range of indices to be used by „Delete“ and „Plot“ can be given.

<u>*Node type*</u>: G, A, F, T for PATRAN interface.

<u>*# Degrees of freedom*</u>: The number of DOF for the nodes must be given.

<u>*Node class:*</u> Class ID for separating between node classes corresponding PATRAN interface can be given.

<u>*Boundary condition flags:*</u> A number consisting of the digits 1-6 must be given (for example the number 1346 stands for the flags 101101).

Local coordinate system

Local coordinate system: The option has to be set if the constraints are referring to a local coordinate system.

ID: The ID of an already defined coordinate system can be given (ID = 0 means global coordinate system). All currently defined coordinate systems are shown within the list box.

Node increment: If the constraints are defined for a local coordinate system which was defined using option „Boundary Curve“ in command **Coordinate System** the same increment as for the definition of the coordinate systems has to be given here.

Node selection

The node selection must be given on which the boundary conditions should be applied.

Plot boundary conditions

Vector length: Max. length of plotted vectors must be given

Components 1-3 / 4-6: Select which components should be plotted. Translations (1-3) or rotations (4-6).

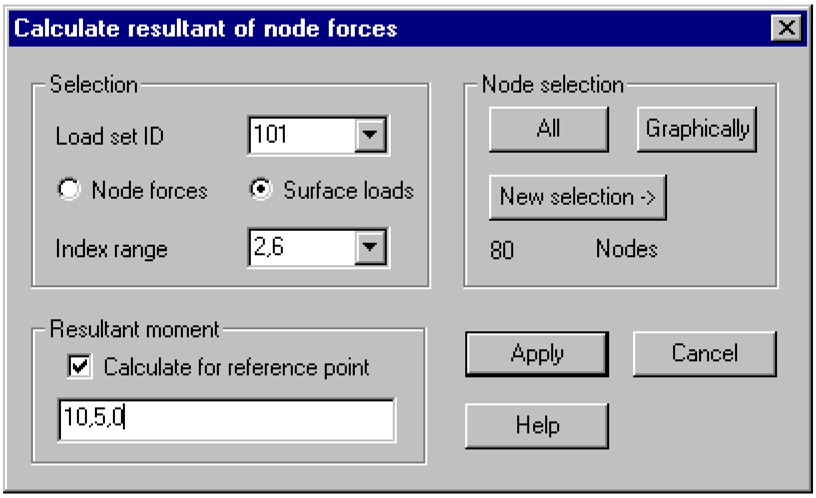
Superimpose: With this option set the previously used layer won't be deleted prior to plotting, so it's possible to do multiple plots using different colors.

Color: Give the color index to use for plotting the vectors.

Resultant: Calculate resultant of node forces

With this command the resultant of node forces and the moment of the forces in reference to a global point may be calculated to check the load definition. Considered are all forces that belong to a given range of load set IDs, to a given range of indices and to a given node selection.

The following dialog shows the available options:



Selection

Load set ID: The list box shows all used load set IDs, a single load set ID or the limits of a range of IDs may be given.

Node forces: This option is to be used, if the resultant of forces should be calculated, that are defined with the command "**Node forces**".

Surface loads: This option is to be used, if the resultant of forces should be calculated, that are defined with the command "**Surface loads**" or "**Surface pressure**".

Index range: The list box shows all for the selected load type used indices. A single index or the limits of a range of indices may be given.

Node selection

Only forces of nodes that are contained within the node selection are considered.

Resultant moment

If the option „Calculate for reference point“ is marked, the resultant moment of the forces in respect to the given reference point is calculated. The reference point must be given in global coordinates.

Load from file: Load data from binary file

This command loads data from a binary file with extension .lqd. The filename has to be given in a file selection box.

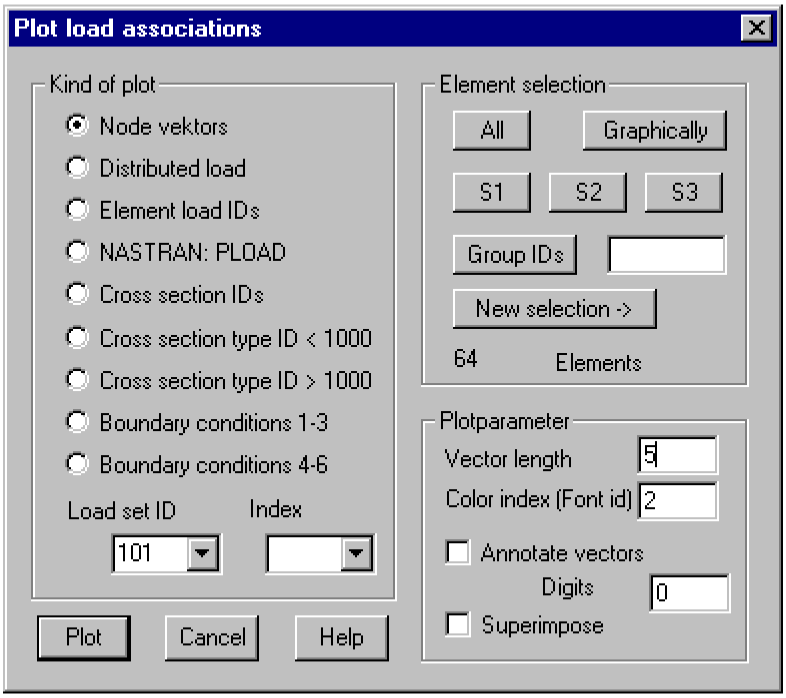
Save to file: Save data in a binary file

This command saves data in a binary file with extension .lqd. The filename has to be given in a file selection box.

Plot: Plot associations

This command is used to checks the associations graphically.

Following Dialog shows the available options:



Kind of plot

Node vectors: The vectors associated with the selected nodes are plotted; this may be forces, moments or displacements.

Distributed load: Plotted are statically equivalent forces belonging to loads defined with the commands "**Distributed load** " and "**Surface pressure**"

Element load IDs: The element load IDs associated with the elements are plotted as a number in the center of element gravity.

NASTRAN: PLOAD: The element surfaces that are loaded with a NASTRAN PLOAD (command "**Element loads**“ with load type ID 10,13,14) are colored.

Cross section ID: The associated cross section ID will be plotted in the center of element gravity.

Cross section type ID < 1000: The associated cross section ID < 1000 will be plotted in the center of element gravity (with NASTRAN interface this are cross-sections with additional data for the element definition).

Cross-section type ID > 1000: The associated cross section ID > 1000 will be plotted in the center of element gravity (with NASTRAN interface this are property records).

Boundary conditions 1-3: First 3 boundary conditions are plotted.

Boundary conditions 4-6: Boundary conditions 4 - 6 are plotted.

Load Set ID: In case a Load Set ID is given only those values for this ID are plotted.

Index: If a range of indices is given only the data belonging to these indices is plotted. The list shows all used indices for the selected type of data.

Plot parameters

Vector length: For forces, moments and boundary conditions this value gives the maximum plotted length measured in drawing units.

Color index (Font ID): The color index used for plotting vectors respectively the font ID for plotting numbers must be given.

Annotate vectors: If this option is set the force vectors are annotated with their value of vector length. In the input field the number of digits used after decimal point must be given.

Superimpose: With this option set the previously used layer won’t be deleted prior to plotting, so it’s possible to do multiple plots using different colors.

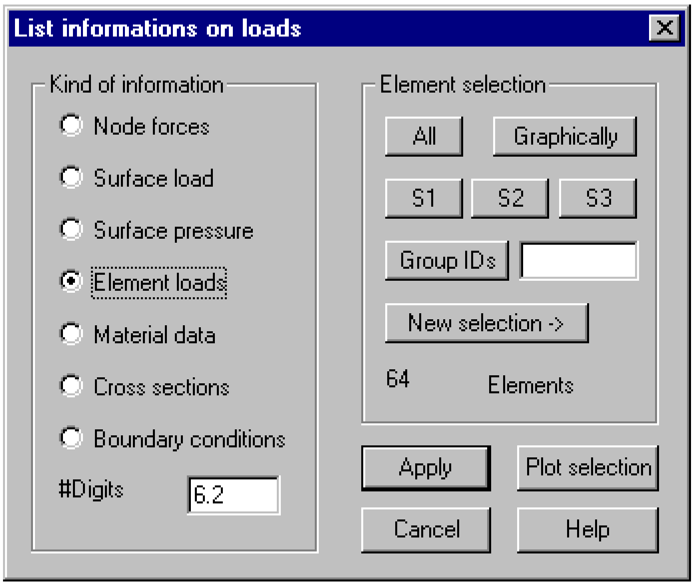
Element selection

The elements have to be selected for which the associations should be plotted.

Information: List associations

With this command several pieces of information about saved associations can be listed. Generating the PATRAN or NASTRAN interface file and printing it can produce a complete listing. The protocol shows up within a popup window.

Following Dialog shows the available options:



Kind of information

Node forces: For each saved load group following values are listed:

Index of load group

LoadSetID

Load components

ID of local coordinate system

Number of associated nodes

Element load: For each saved load group following values are listed:

Index of load group

LoadSetID

Component flag

Largest central component

Node flags

Largest node component

Number of associated elements.

Material data: For each defined material index material type ID and all non zero values are listed

Cross sections: For each defined cross section following values are listed:

Material ID

Element form ID

Configuration ID

Number of cross section values

Number of associated elements

Cross section values

Boundary conditions: For each boundary condition type the number of associated nodes is shown together with ID of local coordinate system

Apply

Raises the popup.

Element selection

The listing can be reduced to selected elements.

Delete all: Delete all data in memory

With this command all load data, constraints, etc., currently in memory is deleted.

Check NASTRAN: Check assignments for NASTRAN interface

Selecting this command causes the following checks to be performed:

Mechanical type ID

For the NASTRAN-interface the association of MAKROS elements to NASTRAN elements is done on the basis of mechanical type IDs assigned to the elements (for example mechanical type ID = 22 results in CQUADR records with property record PSHELL). It is checked whether all elements are assigned a mechanical type ID that is compatible with the geometrical type of the element. Elements that have no or a wrong mechanical type ID are stored as element selection Set 1, so these elements can easily be found graphically.

Property record

It is checked whether all elements is assigned a cross-section with property data that is compatible with the NASTRAN element type (cross section type ID = mechanical type ID +1000). Elements that have no or a wrong cross section assigned to are stored as element selection Set 2 and can easily be found graphically.

Cross-section record

It is checked whether all elements is assigned a cross-section with additional element data that is compatible with the NASTRAN element type (cross section type ID = mechanical type ID). Elements that have no or a wrong cross- section assigned to are stored as element selection Set 3 and can easily be found graphically.

Material

It is checked whether all in the cross-section definitions used material IDs are defined.

Interfaces to FE-Programs

General

NASTRAN	Create or read a NASTRAN input file
PATRAN	Create or read a PATRAN input file
Save ASCII file	Write elements in ASCII format to file
Load ASCII file	Read elements in ASCII format to file
Interface (DLL)	Call interface function from DLL

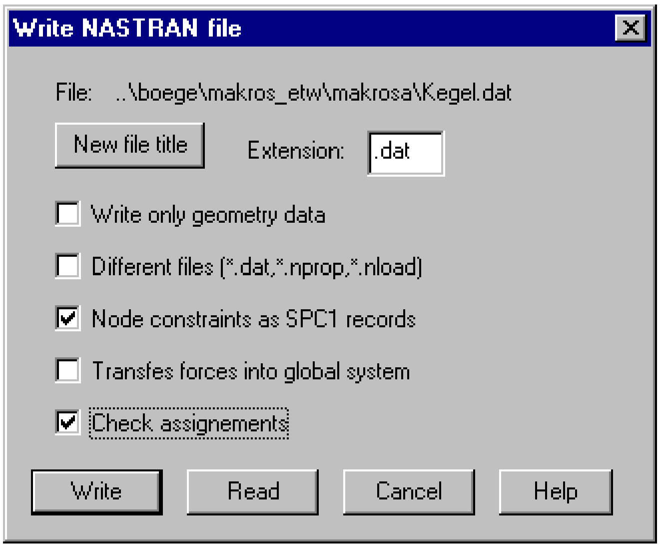
Other interfaces, which are implemented for special applications, are only available, if a file „maka_interface.ini“ with the key word of the interface exists in the folder where makrosa.exe is located.

NASTRAN – Interface

Abbildung 1

With command „**NASTRAN**“ the following NASTRAN-records (see „MSC/NASTRAN Version 70, Quick Reference Guide“) can be written to a file, respectively elements can be read from a NASTRAN file.

Following dialog shows the available options:



New file tile: Default is the current project title; this button allows selecting another title for the NASTRAN file.

Extension: The extension for the NASTRAN file can be given in the input field, default is .dat.

Write only geometry data: With this option set, only nodes, elements and coordinate systems are written to the file.

Different files: With this option set, the geometry data, the property data and the load data are written to different files with extensions .dat, .nprop, .nload.

Node constraints as SPC1 records: With this option set, SPC1 records are generated for single point constraints, else these constraints are stored as parameters CD and PS in the GRID records.

Transfer forces into global coordinates: With this option set, any forces (displacements) that are defined in local coordinate systems are transformed into the global coordinate system before written into the file.

Check assignments: With this option set the following controls as with command „**CheckNASTRAN**“ are executed:

Mechanical type ID: All elements are checked, whether the assigned mechanical type ID is compatible with the geometrical element type (element with wrong assignments are stored in the element selection Set 1).

Property-record: All elements are checked whether a correct property type for the property record is assigned (elements with wrong assignments are stored in the element selection Set 2).

Cross-section: All elements are checked, whether a correct cross-section type with additional element parameters is assigned (elements with wrong assignments are stored in the element selection Set 2).

Material: It is checked, whether all used material IDs are defined.

With button „Write“ a NASTRAN input file is created. The data for the NASTRAN file is read from the files „project.fes“ and „project.lqd“ where „project“ is the current file title of the project. Before reading these files it is asked, whether the files should be updated with data in memory. The answer must be „yes“, if any new defined data is not yet saved to file.

With button „Read“, node data, element data and coordinate systems are read from the given NASTRAN file. Property records and load records are not read.

NASTRAN records

Following is described how the different MAKROS data is translated to NASTRAN records. The interface will be expanded on request.

Node data, single-point constraints

Node data is stored in GRID records. If there are any constraints defined for individual nodes (command “**Constraints**„) these constraints are optionally stored in the GRID record or as SPC1 records.

Elements and element properties (cross-sections)

The translation of MAKROS elements to the different types of NASTRAN elements is done on the basis of assigned mechanical type IDs (mtype).

An element record defines NASTRAN elements and an associated property record in the NASTSRAN input file. The NASTRAN element record stores the element ID, the IDs of connected grid points, the ID of the associated property record and with some element types additional data as for example the thickness of the element.

The specifications for the property record and for additional data if required are defined with the command „**Cross-section**“. Following we use the term „cross-section“ for all this kind of data independent whether it affects the property or the element record. Different kinds of cross-sections are distinguished by a cross-section type ID (qtype). Each cross-section consists of a cross-section ID, a type ID, a material ID and a number of cross-section values. The material number is stored in the property record while the cross-section values are stored in the property record or the element record depending on the used type ID.

The cross-sections that are determined for the property records must be assigned a type ID that is 1000 greater than the mechanical type ID of the corresponding elements (qtype = mtype + 1000) while cross-sections that are determined for the element record must get the same type ID as the corresponding elements (qtype = mtype).

Following table shows which mechanical type ID (mtype, column 1) must be assigned to MAKROS elements to get the NASTRAN elements shown in column 3 with property records shown in column 4. For example assigning mechanical type ID 11 to an element of geometrical type 42 will lead to a NASTRAN record CQUAD8 with property record PLPLANE. In the following table column 2 shows which MAKROS elements may be used for different NASTRAN element records. Column 5 shows the number of cross-section values that must be given for the property record while column 6 shows the number of cross-section values that must be given for additional data in the element record. If the numbers in column 6 are enclosed in brackets the cross-section with type ID qtype = mtype is optionally, for example the thickness of membrane elements can also be given in the cross-sections for the property records if the thickness is the same in all nodes.

Assignment of NASTRAN elements to MAKROS elements

mtype	**MAKROS-** Type	**NASTRAN-** Element-Record	**NASTRAN-** Property-Record	Number of Property values	Number of additional Element values	Remarks
1	2	3	4	5	6	7
11	30	**CTRIA3**	**PLPLANE**	2	[2 o. 6]	6) ,7) ,8)
11	32	**CTRIA6**	″	″	″	″,″
11	40	**CQUAD4**	″	″	″	″,″
11	42	**CQUAD8**	″	″	″	″,″
13	40, 42, 46	**CQUAD**	″	″	–	″
14	30, 32	**CTRIAX**	″	″	–	″
14	40, 42, 46	**CQUADX**	″	″	–	″
15	30, 32	**CTRIAX6**	–	–	1	″, 11)
21	30	**CTRIA3**	**PSHELL**	1 o. 9	[2 o. 6]	6), 7), 9)
21	32	**CTRIA6**	″	″	″	″,″
21	40	**CQUAD4**	″	″	″	″,″
21	42	**CQUAD8**	″	″	″	″,″
22	30	**CTRIAR**	″	″	″	″
22	40	**CQUADR**	″	″	″	″
31	30	**CTRIA3**	**PCOMP**	variable	″	6), 7) , 13), 3)
31	32	**CTRIA6**	″	″	″	″,″
31	40	**CQUAD4**	″	″	″	″,″
31	42	**CQUAD8**	″	″	″	″,″

mtype	**MAKROS-** Type	**NASTRAN-** Element-Record	**NASTRAN-** Property-Record	Number of Property values	Number of additional Element values	Remarks
1	2	3	4	5	6	7
32	30	**CTRIAR**	″	″	″	″
32	40	**CQUADR**	″	″	″	″
41	40	**CSHEAR**	**PSHEAR**	4	–	
60	60, 62	**CPENTA**	**PLSOLID**	1	–	8)
60	70, 72	**CTETRA**	„	„	–	
60	80, 82	**CHEXA**	„	„	–	
61	60, 62	**CPENTA**	**PSOLID**	5	–	22)
61	70, 72	**CTETRA**	„	„	–	
61	80, 82	**CHEXA**	„	„	–	
101	20	**CONROD**	–	–	4	
102	20	**CROD**	**PROD**	4	–	
103	20, 1	**CELAS1**	**PELAS**	3	2	12) , 20)
104	20, 1	**CELAS2**	–	–	5	„ „
105	20, 1	**CDAMP1**	**PDAMP**	1	2	„ „
106	20, 1	**CDAMP2**	–	–	3	„ „
108	20	**CTUBE**	**PTUBE**	4	–	
110	20	**CBAR**	**PBAR**	17	3 o. 6 o. 12	16), 21)
111	20	″	″	„	3 o. 6 o. 12	″, 17)
112	20	″	″	„	2	″, ″, 18)
113	20	″	**PBARL**	variable	3 o. 6 o. 12	″, 10)
114	20	″	″	„	3 o. 6 o. 12	″ , 17)
115	20	″	″	„	2	″, ″ , 18)
120	20	**CBEAM**	**PBEAM**	variable	4 o. 6 o. 12 o. 14	16), 3)
121	20	″	″	„	4 o. 6 o. 12 o. 14	″, 17)
122	20	″	″	„	2	″, ″, 18)
123	20	″	**PBCOMP**	variable	14	″
124	20	″	″	„	14	″, 17)
125	20	″	″	„	2	″, ″, 18)

mtype	MAKROS-Type	NASTRAN-Element-Record	NASTRAN-Property-Record	Number of Property values	Number of additional Element values	Remarks
1	2	3	4	5	6	7
126	20	″	**PBEAML**	variable	14	″, 10)
127	20	″	″	„	14	″, 17)
128	20	″	″	„	2	″, ″, 18)
130	20	**CGAP**	**PGAP**	10	4	
131	20	″	″	″	–	19)
140	20	**CVISC**	**PVISC**	2	–	
150	20	**RBAR**	–	–	4	
201	1	**CONM1**	–	–	22	
202	1	**CONM2**	–	–	11	
210	–	**MPC**	–	–	variable	14)
211	20	″	–	–	Variable	15)

Remarks:

1) To elements that have the same cross-section values can be assigned the same cross-section ID (option Element selection in the command „**Cross-section**“).

2) Cross-section values are written into the property record (after the material ID) respectively into the element record in the same order as given, missing values at the end are set to zero.

3) Values for fields in the NASTRAN records that should remain blank must be given as –99 or as 0. Cross-section values –99 or 0 leave the corresponding field in the NASTRAN record blank.

4) For property records always 6 values for the first line and 8 values for the following lines must be given independent of the type of record. For fields that are blank in the record, the value –99 or 0 must be given in the corresponding cross-section. In case the property record contains no material ID, 7 values must be given for the first line (for example PCOMP, PGAP).

5) For some fields that contain no numerical data a special numerical code is given in the following. If no code is given, these fields must be edited afterwards.

6) mtype = 11-32: Additional values for the element records of CTRIAx and CQUADx are values for THETA, ZOFFS and T1 – T4 (thickness). If no cross-sections for these elements (qtype = 11 – 32) exist, the corresponding fields are left blank. If only 2 values are given in the cross-section, these are used for THETA and ZOFFS, if more values are given, the first 2 values are used for THETA and ZOFFS, remaining values are used for T1 – T4.

7) mtype = 11,21,31: Which NASTRAN element record (CTRIA3 – CQUAD8) is written for mtype = 11, 21, 31 depends on the number of nodes of the given MAKROS element.

8) mtype = 11-50,60: For parameter STR in the property records PLPLANE and PLSOLID the values 0 for „GAUS“ and 1 for „GRID“ must be given.

9) mtype = 21,22: If the cross-section for the property record PSHELL (qtype = 1021, 1022) contains only 1 value, this value is used for parameter T. Parameters MID2 and MID3 are set to MID1 (material ID), the remaining fields are left blank.

10) mtype = 113-115,126-128: For parameter TYPE in records PBARL and PBEAML the numerical code 0 – 18 is translated as follows: 0 = ROD, 1 = TUBE, 2 = L, 3 = I, 4 = CHAN, 5 = T, 6 = BOX, 7 = BAR, 8 = CROSS, 9 = H, 10 = T1, 11 = I1, 12 = CHAN1, 13 = Z, 14 = CHAN2, 15 = T2, 16 = BOX1, 17 = HEXA, 18 = HAT.

11) mtype = 15: For record CTRIAX6 the corresponding cross-section (qtype = 15) must contain the material ID and 1 value for the parameter TH.

12) mtype = 103-106: Additional element data for the records CELAS1, CDAMP1 are the parameters C1, C2, for record CELAS2 the parameters K, C1, C3, GE, S and for record CDAMP2 the parameters B, C1, C2. If the MAKROS element is a point element (gtype = 1), the fields for G2 and C2 are left blank (NASTRAN scalar element).

13) mtype = 31,32: The numerical code for parameter FT in the record PCOMP is as follows: 1 = HILL, 2 = HOFF, 3 = TSAI, 4 = STRN, 0 leaves the field blank. For LAM > 0 „SYM“ and for SOUTi > 0 „YES“ is set.

14) mtype = 210: With mtype = 210 no element but only a cross-section with type ID 210 has to be defined. The values of the cross-section are written into a record of type MPC, the material ID is not used.

14) mtype = 211: With mtype = 211 the nodes of the line element (< = 5 nodes) are use as parameters G1 – G5 in a MPC record, the values for Ci and Ai must be given in a cross-section of type 211.

16) mtype = 110-115,120-128: Cross-sections with additional element data are equally interpreted for element types CBAR and CBEAM. With CBAR the value for field 9 and for parameters SA and SB must be given as 0 or –99. If only 3 (4) values are given, these are used for X1, X2, X3, (BIT), with 6 values given also PA, PB are set and with 12 (14) values given the last values are used for parameters W1A – W3B and SA and SB.

17) mtype = 111,112,114,115,121,122,124,125,127,128: The alternative format of records CBAR respectively CBEAM is used where the 3. node of the element is used for parameter CO, values for X1- X3 in the cross-section are not used.

18) mtype = 112,115,122,125,128: The vector from node 4 towards node 1 of the element is used for the parameters W1A – W3A, and the vector from node 5 towards node 2 of the element is used for the parameters W1B – W3B (the fields remain blank, if nodes 4 respectively 5 are zero). In the corresponding cross-section only values for parameters PA and PB must be given.

19) mtype = 131: The alternative format of record CGAP is used where 3. node of the element is used as a parameter GO, CID is set to zero.

20) mtype = 103-106: Only 1 property cross-section is stored per record.

21) mtyp = 110: To CBAR elements an additional cross-section with ID = 2110 can be assigned that contains 4-6 values. This cross-section is stored as CBARAO record. For parameter SCALE 1 for LE and 2 for FR has to be given. With SCALE = 1 and 4 cross-section values these are written as SCALE, NTPS, X1 and DELTAX, else the given values are written as SCALE and X1-X6.

22) mtyp = 61: For FCTN not zero FLUID is set..

Material data

Material data is defined with command „**Material**“. Each record contains a material ID, a material type ID and a variable number of material values. The material type ID (matype) is used to distinguish different NASTRAN material records as shown in following table:

matype	NASTRAN	matype	NASTRAN	matype	NASTRAN
1	MAT1	2	MAT2	3	MAT3
4	MAT4	5	MAT5	8	MAT8
9	MAT9	10	MAT10	11	MATHP

The material values are written into the NASTRAN record in the same order as given. For the first line 7 values and for following lines 8 values must be given independent of the record type and the number of values in the lines of the record. For fields that are blank in the record, the value 0 or –99 must be given.

Node forces, moments or displacements

Node vectors are defined with the command „**Node forces**” With a type ID (ltype) different types of vectors are distinguished as shown in following table:

ltype	Vector type	NASTRAN-Record
0	Force vector	FORCE
1	Moment vector	MOMENT
2	Displacement vector	SPC
3	Rotation vector	SPC
4	Displacement vector	SPCD
5	Rotation vector	SPCD

The vectors may be defined in a local coordinate system. In the NASTRAN dialog box can be specified that vectors defined in local coordinate systems should be transformed to global coordinates before written to the NASTRAN file. Written into the NASTRAN records are the load set ID, the node ID, the ID of the local coordinate system, the scale factor and 3 vector components. With command „**Surface loads**“ distributed loads can be defined that are immediately converted to statically equivalent node forces. These forces are also written into records of type FORCE.

For ltype = 2 and 3 records of type SPC and for ltype = 4 and 5 records of type SPCD are written, that means the vectors are interpreted as enforced displacement vectors.

Single point constraints

Single point constraints are defined using command „**Constraints**“, they are alternatively written into the GRID records or SPC1 records. Enforced displacements (NASTRAN records SPC and SPCD) can be defined with command „**Node forces**” (see above). Multi point constraints are defined as a cross-section with type ID 210 or 211.

Element loads

Element loads are defined with command „**Element loads**“. With a load type ID (latype) different NASTRAN records are distinguished as shown in following table:

latype	NASTRAN-Record	Number of values
10	PLOAD	1
11	PLOAD1	6
12	PLOAD2	1
14	PLOAD4	4 (10)

In the dialog only the following data must be given: the index, the load set ID, the load type ID and the load values, these must be given in the input field for central load components (EFLAG = 1, GFLAG = 0). The other input fields are not used for NASTRAN data. As load specifications the following values must be given for the NASTRAN records:

latype = 10: 1 Value for P

= 11: Codes for TYPE, SCALE and 4 values for X1, P1, X2, P2

= 12: 1 Value for P

= 14: Values for P1 – P4 and with solid elements values for G1, G3, CID, N1 – N3

For PLOAD the corner nodes of the selected elements are used. The sign of P must be chosen to correspond to the orientation of the element. With solid elements these surfaces are loaded whose corner nodes are all contained in the given node selection.

With PLOAD1 first a code for the parameter TYPE must be given. The code is 1 – 12 for the following character strings: FX, FY, FZ, FXE, FYE, FZE, MX, MY, MZ, MXE, MYE, MZE. Next the code for parameter SCALE must be given, the numbers 1 – 4 stand for LE, FR, LEPR, FRPR.

If with latype = 14 a node selection is given, the nodes for the parameters G1 and G3 of the record PLOAD4 are automatically determined for solid elements, in this case given values for G1 and G3 must be zero.

PATRAN Neutral File

Command **PATRAN** creates a neutral PATRAN file with currently saved data values of the Finite element model (see PATRAN Plus User Manual, chapter 29). The filename is selected from a file selection dialog.

PATRAN differentiates data cards (in the following called packet type) with a 2-digit type ID. Each data type starts with a header card in the format (I2, 8I8) with the following information:

IT = Data type

ID = Identification number

IV = Additional ID

KC = Number of following cards for this data type
N1-N5 = Integer values

Such a header card follow KC cards with values related to the data type.

In the following it will be stated how the conversion between MAKROS data and PNF data is done.

Packet type 25: Title Card

The following card gives the file name under witch the data is stored on hard disk.

Packet type 26: Summary Data

The header card contains the following data:
N1 = Number of nodes
N2 = Number of elements
N3 = Number of materials
N4 = Number of element properties
N5 = Number of coordinate frames

The following card contains the date and time of file creation

Packet type 01: Node data

This card saves nodal coordinates and all defined boundary conditions specified by the command Constraints.

First card saves external node ID. 2^{nd} card specifies global cartesian x,y,z-coordinates of this node.

The 3rd card gives parameters ICF = 0, GTYP, NDF, CONFIG, CID and PSPC just as specified by the command **Constraints.**

Packet type 02: Element data

This card saves the element related parameters.

The 1st card holds external element ID, shape ID for this element (2 = bar, 3 = tri, 4 = quad, 5 = tet, 7 = wedge, 8 = hex), the number of associated data values (N1) and if necessary node ID of node in xy-plane (the 3rd node) in case of bar elements.

The 2nd card holds the number of element nodes corresponding to element type. CONFIG holds the group ID and PID holds cross section ID associated with this element. CEID is set to 0.

Next cards hold node IDs for this element.

Packet type 03: Material Properties

This card contains material data specified by the command **Material**.

The 1st card saves material ID and type ID for this material. N1 will be set to 0. Next 20 cards hold 96 material values in the same order they were entered. Missing values are set to 0.

Packet type 04: Element properties

This card contains cross section properties defined by the command **Cross section**.

The 1st card holds cross-section ID and material ID. For each element type and for each material one cross section type must be defined and associated to an element. N1 and N2 define element shape and the number of element nodes just as for package type 02. N3 holds the given configuration flag, N4 holds the number of saved data values.

Packet type 05: Coordinate systems

This card contains the parameters for local coordinate systems.

The 1st card holds external ID of the system and a type ID (1 = cartesian, 2 = cylindrical, 3 = spherical).

Additional cards hold the coordinates of 3 points defining this coordinate system and 9 values defining the transformation matrix.

Packet type 06: Distributed element loads

This card contains distributed element loads defined by the command **Element load.**

The 1st card holds element ID and load set ID.

The 2nd card holds the given parameter values for LTYPE (0 = edge load, 1 = surface load), EFLAG, GFLAG, ICOMP, NODE, NFE (ID for edge respectively surface for solids).

Additional cards hold the given load components.

Packet type 07: Nodal loads

This card contains the nodal loads defined by the command **Node forces**.

The 1st card holds node ID and load set ID.

The 2nd card holds the ID of the local coordinate system and load component flag.

Additional cards hold load components if load component flags = 1.

Write ASCII files

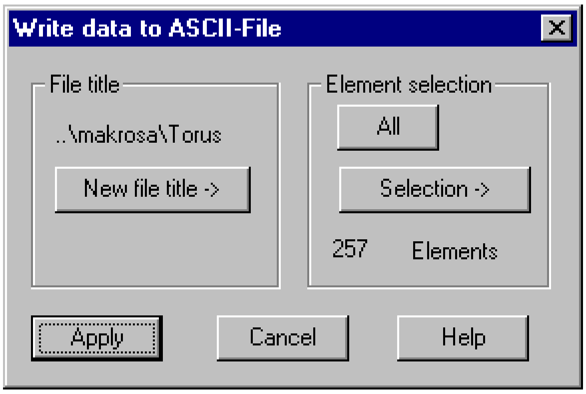
Write ASCII file: Save elements to file in ASCII format

All or selected elements can be saved to a text file. Nodes and elements are saved into different files with extensions .efp (nodes) and .efe (elements), having the same basis name.

Node file (*.efp) contains external node ID and three coordinate values for each node.

Element file (*.efe) contains external element ID, geometrical type ID, group ID, mechanical type ID and external node IDs of the element nodes.

Following dialog shows the available options:



File title

The current title of the project is shown which could be used as the basis name for the files.

New file title: A file selection box pops up to select a new title

Element selection

All: Save all elements to file

Selection: Dialog box for the element selection popes up. Only selected elements and associated nodes are saved to file.

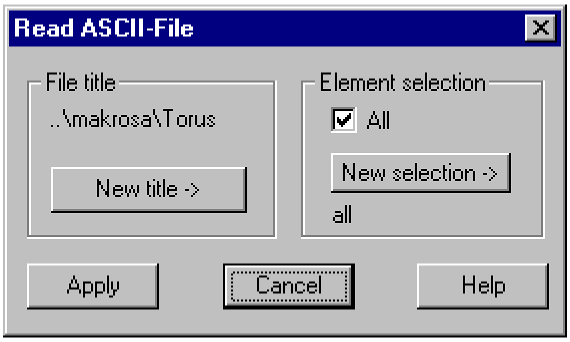
Read ASCII files

Read ASCII file: Load elements from file in ASCII format

All or selected elements can be read from file. Nodes and elements are read from different files with extensions .efp (nodes) and .efe (elements), having the same basis name. Node file (*.efp) must contain the external node ID and three coordinate values for each node. Element file (*.efe) must contain external element ID, geometrical type ID, group ID, mechanical type ID, and external node IDs of the element nodes. The number of external nodes depends on the element type. Optionally in the first line of the element file a number m may be given, in this case m numbers are read for each element independent of the element type, if the number of node Ids is less than m-4, additional values must be given as zeros.

Note: If the node file contains nodes that are not used as element nodes, these nodes get a single node element assigned to.

Following dialog shows the available options:



File title

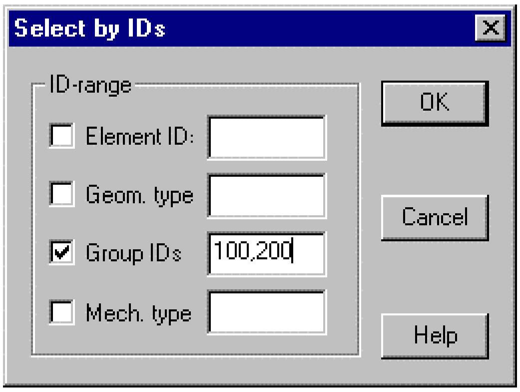
Current title of the project is shown which could be used as the basis name for the files.

<u>*New title:*</u> A file selection box pops up to select a new title

Element selection

<u>*All:*</u> Read all elements from file

<u>*New selection*</u>: The following dialog pops up where elements can be selected. Only elements which meet the given criterions (IDs within given ranges) will be read from file.

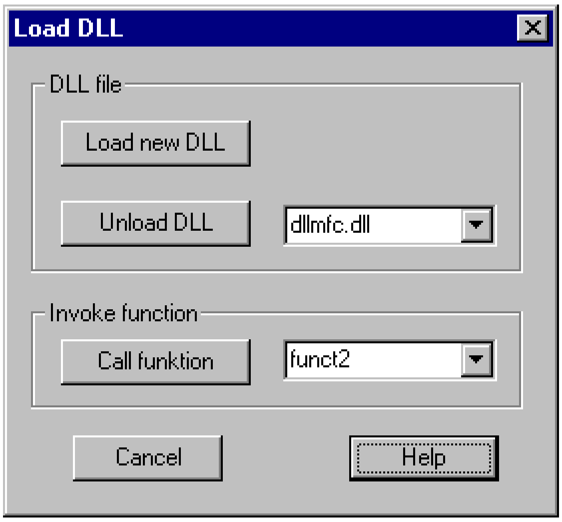


Call interface function from DLL

Interface (DLL): Calling own interface functions from a DLL

This command allows calling own interface functions. The functions must be provided in one or more DLL files.

Following dialog shows the available options:



DLL file

Load new DLL: Clicking this button, a file selection box pops up, where the file of the DLL to be loaded must be selected.

Delete DLL: All loaded DLLs are show in the list box. After selecting a DLL it can be deleted by clicking the button.

DLL function

Call function: A function name must be given in the input field. Clicking the button, this function will be called. If the function is called the first time, the name of the DLL where the file is contained in must also be selected in the list of DLL names.

The list shows all the functions that have previously called; these functions can be called several times.

When the command is invoked the first time, it is checked, whether there exists a file „maka_interface.ini“ in the “bin” folder of MAKROS that contains some lines of following type:

#dll kf function path

„kf“ is the type ID of the function, „function“ is the function name of an exported function in the DLL and path is the path of the DLL, the function is contained in.

For example:
#dll 0 funct1 G:\boege\makrosa\testdll\dll1\debug\dll1.dll
#dll 0 funct2 G:\boege\makrosa\testdll\dllmfc\debug\dllmfc.dll
#dll 1 testInterface G:\boege\makrosa\testdll\interf\debug\interface.dll

If lines of this type are found, the DLLs are automatically loaded and the entry points of the functions are searched and stored. The DLL and function names are shown in the list boxes of the dialog window.

Prototype of functions

Functions of this type (type ID 1) must have the following prototype:

extern "C"
void Interface(void ***pCWnd**, StructData ***data**, char ***file**, int **kz**, int **kzf**, void ***p**);

The arguments have the following meaning:

pWnd: pointer to the protocol window of MAKROS, this window should be used as parent, if dialog windows are used in the function,

data: Pointer to a structure, that contains all project data that is needed for the interface file. The structure “StructData” is fully described in the file "interface.h“,

file: title of the file to be created,

kz: not used with this kind of functions,

kzf: ID of the file type, must be 1 with this type of functions,

p: pointer to a structure not used with this type of function.

If a function of this type is called, first it is asked, whether all data is saved to disk, because the data that is supplied to the function is read from the binary files of the current project. Then the file title to be used for the interface file must be given in a file selection window, this file title is used as parameter 3 of the function.

A template DLL of this type is contained in the folder dll/interface on the CD.

Development of special Interface programs

The folder “interface” on the installation CD respectively the Zip file contains the following source files that can be used as a model for developing special interface programs, also is it possible to expend the given NASTRAN or PATRAN interfaces:

nastran_interface.cpp: the main-function of the NASTRAN interface program „nastran.exe“.

patran_interface.cpp: the main-function of the PATRAN interface program „patran.exe“.

ladedatei.cpp: function for reading the binary files „project.fes“ and „project.lqd“.

getData.cpp: a function that provides all project data in a special data structure.

nastran.cpp: function that creates the NASTRAN input file.

patran.cpp: function that creates the PATRAN input file.

The function „getStructData“ reads the binary files of the current project from hard disk and provides these data in a data structure for further processing. The data structure is described in the file “interface.h”. The functions “nastran” respectively “patran” write these data corresponding to the format of these interfaces to text files. These source programs can be used as model for the development of other interfaces.

The command Interface (DLL) allows to call own interface programs from a DLL:

Visualize calculation results (Post processing)

Overview

MAKROS provides commands for visualizing results of finite element calculations in several graphical forms.

Following data types can be visualized:

Scalar fields that are node or element related.

Graphics are done by the command **Scalar field**. Distributions of scalar fields can be plotted as iso lines, iso surfaces, column charts or xy-plots.

Node or element related vector fields.

The graphics are done by the command **Vector field**. Vectors can be plotted as a direction. In case of deformation vectors, it's also possible to plot the deformed structure or to do a harmoniously simulation.

Element related vector crosses.

The graphics are also done by the command **Vector field**. The 2 vectors building the cross can be plotted with different colors.

Within menu group **Post processing** following commands are available:

Load post data	Load data for post processing from text files
Visualize	Specify the kind of visualization for scalar fields
Scalar values	Select range of scalar values to be displayed
Node ↔ Element values	Conversion between node and element values
Color definition	Define color values for color indices
Load case	Define combinations of load cases
Element selection	Select elements for structure or scalar/vector field plot
Individuals	Show scalar data for individual nodes/elements
Color bar	Define position of color bar or define additional text
Array Plot	3D Plot or XY Plot of array data
Vector plot	Set parameters for following plots of vector fields
Deformation	Plot deformed structure
Vibration	Show a dynamically simulation

Providing data for post processing

Data for post processing is provided within text files and read with command **Load post data**. The file must contain node or element IDs and node respectively element related values for a single or for multiple load cases.

A type ID in the first line of the file differentiates between following types of data:

ktyp = 0: 1 scalar value is given for each node
= 1: 3 vector components are given for each node
= 2: 1 scalar value is given for each element
= 3: mw scalar values are given for each element: scalar values are assigned to the element nodes.
= 4: mw scalar values are given for each element: 1 scalar value for each surface of volume elements

= 5: 3 vector components are given for each element

= 7: 1 vector cross is given for each element

= 8: mw scalar values are given for each node (mw load cases)

= 9: mw scalar values are given for each element (mw load cases)

= 11: mw vectors are given for each node (mw load cases)

= 12: mw vectors are given for each element (mw load cases)

= 14: mw vector crosses are given for each element (mw load cases)

= 20: mw scalar values are given for each element: 1 value for each integration point

= 21: mw vectors are given for each element: 1 vector for each integration point

= 22: mw vector crosses are given for each integration point

= 30: 1 scalar value is given for each beam element

= 31: 2 scalar values are given for each beam element

= 32: mw scalar values are given for each beam element

= 33: 2 * mw scalar values are given for each beam element

The data files must have the following format:

First line: 7 [3] integers for ktyp, mw, lc, mip [, nl1, nl2, nl3]

ktyp defines the type of data contained in the file.

mw is the number of scalar or vector values belonging to each node or element. If the number of values is not the same for all elements, the maximum number must be given for mw and missing values for some elements or nodes have to be added as zeros. This is for example the case when scalar values are given in integration points and when the structure is build of quadrilateral and triangular elements where the number of integration points is different for the two types of elements.

lc is the ID of a load case for separating multiple load cases.

mip is the number of integration points for one direction (used only for ktyp = 20 – 22).

nl1 is the number of comment lines before the first data line. For nl1 > 0 nl1 lines are skipped after the format line.

nl2 is the number of continuous data lines. For nl3 = 0 nl2 has no effect.

nl3 is the number of comment lines that follow each block of nl2 data lines. For nl3 > 0 nl3 lines are skipped after nl2 data lines (see demo file „nastran2.stress).

Second line: textual data

Given text in this line is used as a description of this load case, which can be plotted during graphical output. This line can be left blank.

Third line: format

This line specifies a FORTRAN like format for the following data lines. Leaving this line blank causes the file to be read in a free format. Reading in free format can only be used when all data lines contain exactly the number of values expected by the file type, and no comment lines are contained in the input file.

The format (I5, 5X, 3F12.0) for example will read node ID and first 3 displacement components from the following input line which contains additional data:

```
 8137    G -7.833E-07 -1.608E-06      0.0       0.0       0.0 -9.991E-08
```

If a data set for one element or one node continues over several lines the format must contain a / for each new line. For example using the format (I10, F15.0, 30X, F15.0 / 25X, F15.0, 15X, F15.0) will read the ID and 1. and 4. value from the first line and 2. and 4. value from the second line of the following data set:

```
    1728  7.390112E-01 -1.320112E-01 -5.322112E-02  0.590112E-01 -1.053304E+01
         -7.439936E-09  2.409936E-03  5.489936E-01 -7.439936E-00
```

The format

(1X,I11////115X,F15.0///115X,F15.0///115X,F15.0///115X,F15.0///115X,F15.0///115X,F15.0///115X,F15.0///115X,F15.0//)

is used to read the Von-Mises-Stresses on the 8 corner nodes of a NASTRAN element CHEXA (see demo file „nastran2.stress)

Hint: No FORTRAN routines are used to read the input lines, a given format is interpreted. Only following specifications are allowed: m**I**n for integers, m**F**n.0 for real values and n**X** for fields that should not be read; where m is an optional multiplier and n the field width. If an error happens when reading an item no further reading is done. If the input file contains comment line, the params nl1-nl3 must be given appropriate.

Additional lines: data values for nodes respectively elements

Each line must start with a node or an element ID followed by mw scalar respectively vector values. Node and element IDs are taken as external IDs, the data values are assigned corresponding to these IDs. If no data values are given for some nodes or elements, corresponding values are set to zero.

With data types 7,14 and 22 (vector crosses) the first two data values are interpreted as length of the two orthogonal vectors while the third data value is taken as an angle (degree) in the element plane.

Files with data types 0-7 and 20-33 are interpreted and saved as a single load case while files with type 8-14 are interpreted as mw load cases. For example it’s possible to save the results of several following steps of a nonlinear calculation or node displacements of following time steps within a single input file.

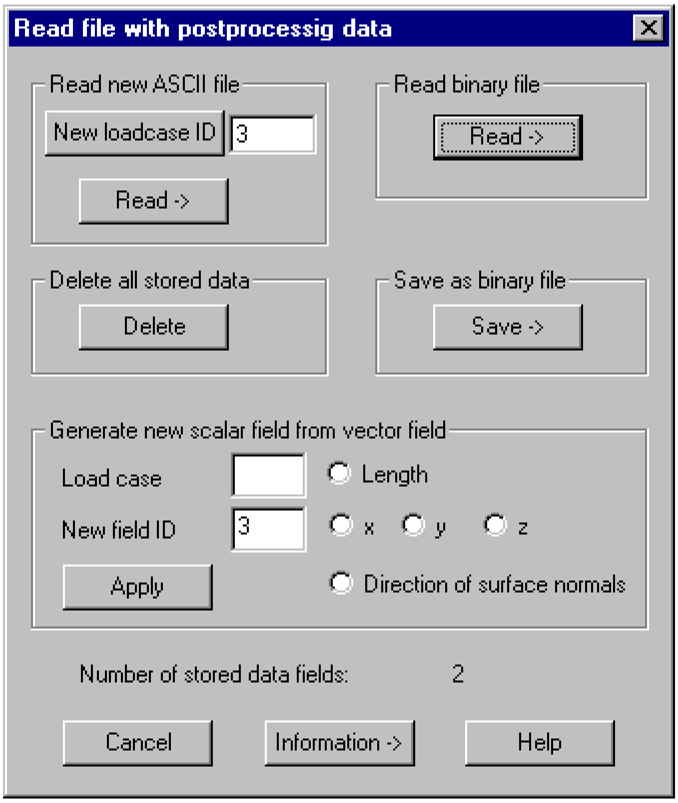
With ktyp = 20 –22 mip is the number of integration points for one direction. Permissible values are mip = 2, 3, 4, 5, 6 with quadrilateral elements and mip = 2, 3, 4 with triangular elements. The number of integration points for one element is mip * mip with quadrilateral and 3, 7, 13 with triangular elements when mip = 2, 3, 4. The position of the integration points is assumed to be as with gauss integration. The ordering is shown in following picture for mip = 4 (see K. J. Bathe: Finite element methods). Integration points are only realized for quadrilateral and triangular elements.

If scalar values are given in integration points, the values for the nodes of the elements are calculated by extrapolation respectively interpolation for additional nodes in the inner of curved elements. If nodes on element edges belong to more than one element, optionally the smallest, the largest or the mean value of the values belonging to the different elements can be used for these nodes. If with ktyp = 3 the number of given scalar values is less than the maximum number of element nodes, it is assumed, that only scalar values for corner nodes of the elements are given, the scalar values for nodes on the edges are in this case calculated by interpolation (see the demo example “nastran2.dem”).

Load post data: Load post processing data

It’s possible to load multiple data sets of the same or a different type.

Following dialog shows the available options:



Read new text file

Read: After pressing this button a file selection box pops up to select the data filename to be read from.

New load case ID: Pressing this button, the next free load case ID will be set in the input field. If a load case ID > 0 is given, this will be used for the newly read data, else the load case ID will be taken from the first line of the file.

Delete all data in memory

Pressing button “Delete” all currently loaded data will be deleted.

Save as binary file

Pressing button “Save” all post processing data currently held in memory is saved to a binary file. The filename is selected from a file selection dialog. File extension will be .pos.

Read binary file

This button allows to loads a previously saved binary file.

Generate new scalar field from vector field

This option makes it possible to save the vector length or a single vector component from the vectors of a given vector field as a new scalar field.

Load case ID: The ID of the vector field to be translated to a scalar field must be given.

New field ID: This input field specifies the new load case ID for the generated scalar field. Next available ID is automatically set.

Length: With this option set the length of the vectors is used as scalar data.

x, y, z: This options select a vector component to be used as scalar data.

Direction of surface normals: This option uses the projection of the given vectors along the direction of the surface normal vectors of the elements as a scalar value.

Apply: After specifying the options this button does the calculation and saving of the scalar values.

Information

This button protocols all currently used data sets (load case, type, smallest and largest value) within a protocol window.

Plot scalar fields

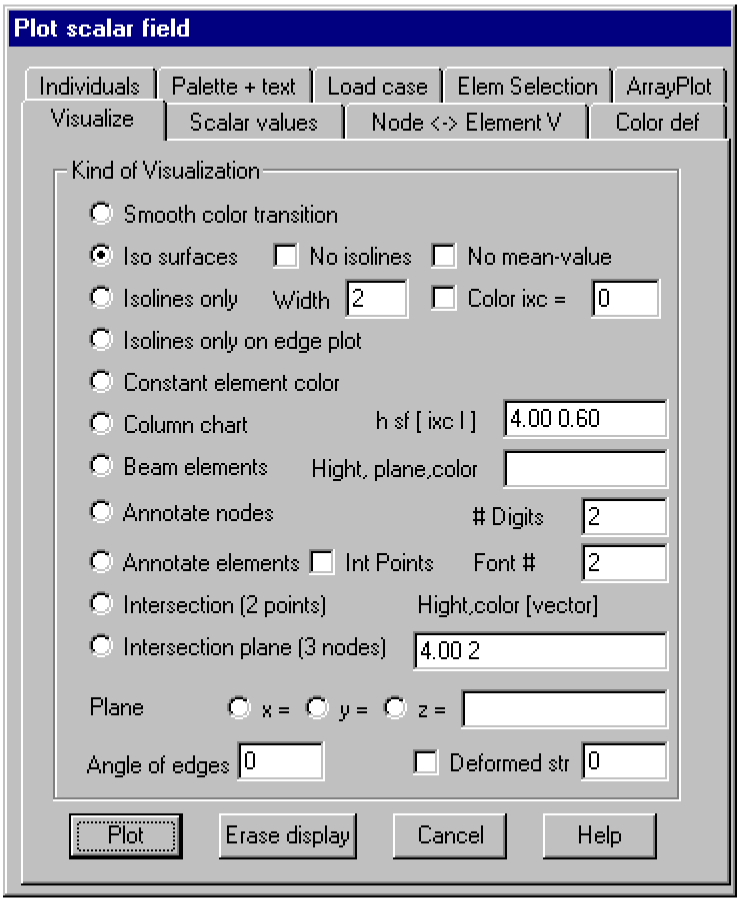
After starting the command **Scalar field** a dialog with several property sheets pops up. This dialog remains active until it's explicitly closed by „Cancel“. It can freely be switched between these sheets.

For the visualization of the distribution of the values of a given scalar field, elements with additional nodes on the edges (for example curved elements) and elements that have scalar values given in integration points are approximated by triangular surfaces. The number of such triangles is 32 with quadrilateral and 16 with triangular elements. Scalar values for the additional nodes used by this approximation are calculated by interpolation using all given scalar values on element nodes and integration points.

Visualize: Specify the kind of visualization for scalar fields

On this property page you can specify the kind of visualization to be used for subsequent plots. After setting the appropriate option button „Plot“ starts the graphics. Button „Erase display“ can be used to erase all display lists.

Following dialog shows the available options:



Smooth color transition

Color values are assigned to the nodes of the elements corresponding to the scalar values belonging to the nodes. Among these points a continuous color transition is used. The assignment of color indices to scalar data is done within property page **Scalar values**.

Iso surfaces

At first, isolines are calculated for individual scalar values. Next the area between to adjacent isolines is filled with a constant color. Optionally isolines can also be plotted. The calculation of isolines is done by linear interpolation using the scalar values calculated for the nodes of the triangles that approximate the elements. If nodes belong to more than one element, the median of the values resulting from the different elements is calculated. If option “No median” is set, no median values are calculated, so there is no continuously transition on element edges.

Iso lines only

No color filling between iso lines is done. Iso lines are plotted using a color corresponding to the scalar value of the iso line. If option “Color ixc” is checked, all iso lines are plotted with the given color in the input field. In the input field for width the width for plotting the iso lines may be given. With this option a hidden line plot is done.

Iso lines only on edge plot

The difference of this option to the option before is, that no hidden line plot is done; only iso lines and element edges are plotted.

Constant element color

With this option set all element surfaces are filled with a constant color. The color is determined by the associated scalar value of the element.

Column chart

Perpendicular to the element surfaces columns are plotted with heights proportional to the elements scalar value. In the input field height h for the largest column, scaling factor sf for the columns and optionally a constant color index ixc and a flag l (shading flag) must be given. The scaling factor determines the ratio between the width of the columns and the edge length of the elements. If there is no color given the color is determined by the scalar value of the element. For a flag l > 0 the columns are shaded.

Beam elements

This option is intended for bar and beam elements with scalar fields of type ktyp = 30 – 33. Perpendicular to the axis of the element the distribution of the scalar values is plotted as a colored bar in the given plane. In the input field, following parameters must be given: Plotted height for the maximal scalar value, 3 components of a vector lying on the plane in which the distribution should be plotted and the color to be used for the plot. The vector must not be parallel to a beam axis. Several plots are superimposed.

Annotate nodes

With this option all nodes contained within the current selection are annotated with associated scalar values. The number of digits following the decimal point and the font index (1-3) for text height and color must be given.

Annotate elements

This option shows the associated scalar value for the elements at the center of gravity. Further options are the same as with option “Annotate nodes”. If scalar values are given for integration points, option “Integr point” can be checked; in this case the integration points are annotated with corresponding scalar values.

Intersection (2 points)

With this option 2 points defining the end points of an intersection line must be graphically selected within the current view after pressing button „Plot“. On each intersection line all scalar values at the intersection of the line with element edges are calculated using linear interpolation. The scalar values are plotted along the intersection line. Within the input field maximum height for scalar distribution, color index for filling and optionally 3 components of a direction vector for the surface can be given. In case there is no direction vector given the plot is done perpendicular to the elements surface. Several intersection lines can be selected sequentially until right mouse button is pressed.

Intersection plane (3 points)

By selecting 3 points after pressing button "Plot", a spatial intersection plane is generated. Scalar values are calculated at all intersections between this plane and element edges. Scalar values are plotted within this intersection plane. Additional parameters to be specified are the same as with option "Intersection".

Plane x, y, z

This options do intersections parallel to global coordinate system planes. Within the input field several values for x,y or z-coordinates of these intersection planes can be given.

Angle of edges

If an angle > 0 (degrees) is given only these element edges are plotted whose angles between the normal vectors of adjacent element surfaces are greater than this value.

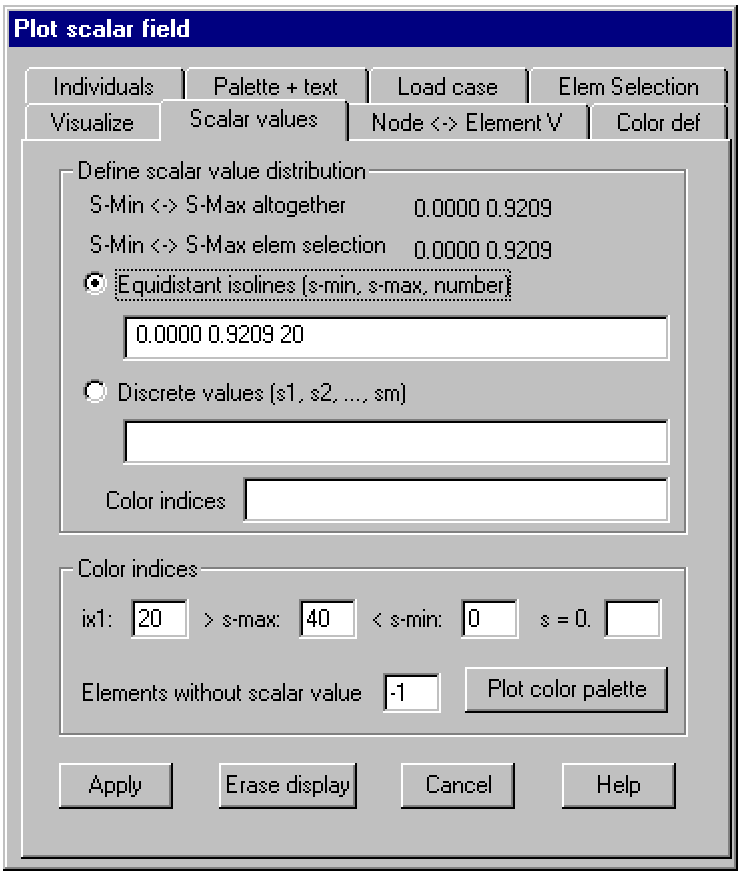
Deformed structure

With this option set the plot is done within the deformed element structure. This is only applicable if there is also a vector field with node displacements loaded and activated by the command **Vector field**. The maximum length for displaying a displacement must be given.

Scalar values: Select range of scalar values to be displayed

This property page defines which scalar values should be used for calculating isolines and which color indices should be used when plotting isobars. The smallest and largest scalar value for all elements and for the elements contained in the selection for the scalar field plot are shown in the dialog.

Following dialog shows the available options:



Equidistant isolines (s-min, s-max, number)

With this option the smallest and the largest value used for a sequence of equidistant isolines must be given together with the number of steps between.

Discrete values (s1, s2, ..,sm)

When selecting this option up to 30 values for individual isolines can be given. Additionally within the second input field color indices for color filling between these isolines can be given. If not enough color indices are given subsequent indices are incremented by 1 (beginning with ix1 if no indices at all are given).

Color indices

The following color indices have to be given:

ix1: Color index for the first isoline starting at s-min. This index will be incremented by one for subsequent isolines.

> s-max: Color index used by isolines greater than s-max respectively sm.

< s-min: Color index used by isolines lower than s-min respectively s1.

Elements without scalar value: Color index for those elements whose scalar values are not to be plotted. Giving index 0 plots with background color, giving index –1 plots the elements with colors assigned to the elements.

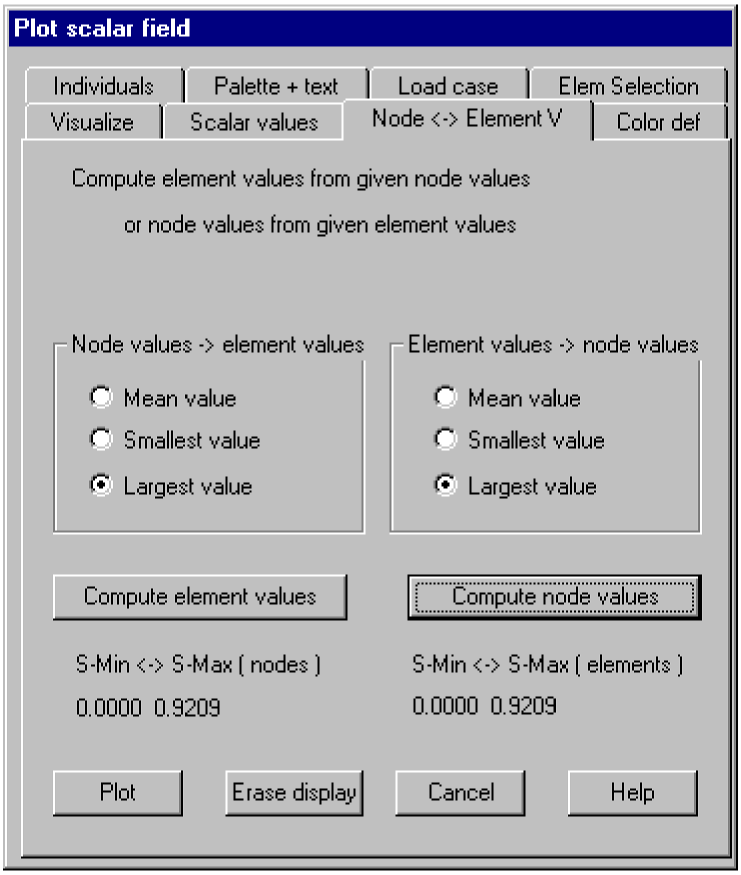
Plot color palette: Currently set color values are plotted as a color palette.

Node ↔ Element values: Conversion between node and element related scalar values

Depending on the kind of the visualization node or element related scalar values are needed. This property page defines how node related values are calculated when element related values are given and conversely.

If scalar values are given at the integration points of the elements, values for the element nodes and for additional nodes used for element approximation are calculated by interpolation respectively extrapolation using all given scalar values. If only one scalar value for the element is needed (for example plot of columns or plot with constant element color) a mean value is calculated using the factors of gauss integration.

Following dialog shows the available options:



Node values → Element values

Mean value: The mean of all values given at the nodes of the element is used for this element.

Smallest value: The smallest of all values given at the nodes is used.

Largest value: The largest of all values given at the nodes is used.

Element values → Node values

Mean value: For each node of the structure the mean of the scalar values from all elements belonging to this node is used.

Smallest value: The smallest value from all associated elements is used.

Largest value: The largest value from all associated elements is use.

Compute element values

Clicking his button the scalar values for the elements are newly computed (available only if node values are given).

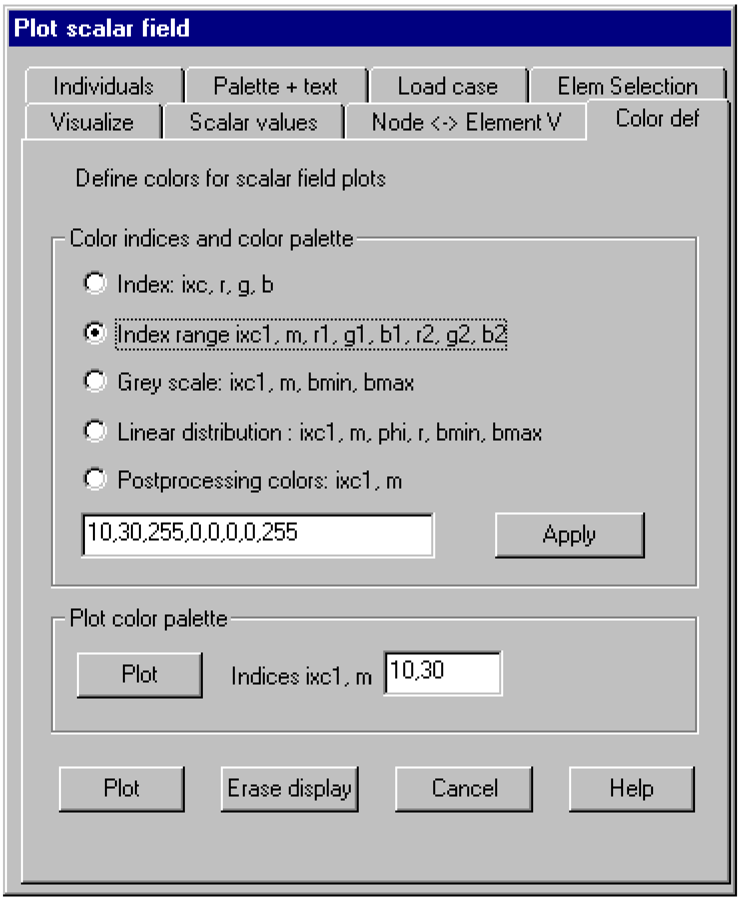
Compute node values

Clicking this button the scalar values for the nodes are newly computed (available only if element values are given).

Color definition: Define color values for color indices

This property page redefines colors associated to color indices. RGB-values are to be given in the range of 0 – 255.

Following dialog shows the available options:



Index: ixc, r, g, b

The color index and associated RGB-values must be given for a single index

Index range: ixc1, m, r1, g1, b1, r2, g2, b2

The smallest color index ixc1 and the number m of indices for which new values should be defined together with RGB-values for the smallest and the largest color index must be given. Color values for all indices between are calculated using linear interpolation.

Grey scale: ixc1, m, bmin, bmax

The smallest color index ixc1 and the number m of color indices together with the brightness value for the smallest and the largest index must be given in %. Indices between are calculated using linear interpolation.

Linear: ixc1, m, phi, bmin, bmax

The smallest color index ixc1, the number of indices m, the color value phi, the smallest brightness bmin and the largest brightness bmax must be given. Color is defined by an angle phi (0-360), a radius r (0-100) and a brightness b (0-100) of a color cone (HLS-system).

Following colors are assigned to angles: 0° = blue, 60° = magenta, 120° = red, 180° = yellow, 240° = green, 300° = cyan, 360° = blue.

Post processing colors: ixc1, m

The smallest color index ixc1 and the number m of following indices must be given. Between these indices a smooth color transition of the colors blue, green, yellow, red and magenta will be created.

Apply

After setting the options this button does the calculation of the colors and updates the color lookup tables.

Plot color palette

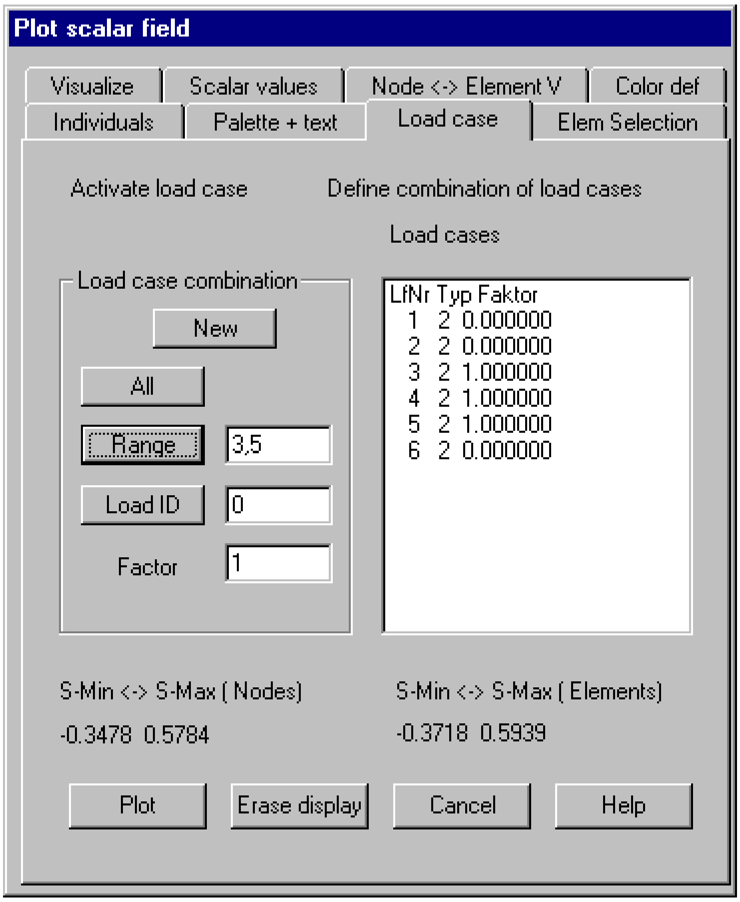
This button plots current color settings as a color bar for indices ixc1 until ixc1+m-1.

Button „Plot“ updates the current scalar data plot with changed color settings.

Load case: Define combinations of load cases

In case there is more than one load case loaded in memory this property page allows the selection of the active load case. It’s also possible to combine several scalar fields of the same type to a new scalar field. The list shows the currently loaded scalar fields together with type and scaling factor. Scalar fields are identified by a load case ID. The type ID identifies the type of the scalar field. For each load case a scaling factor must be given which is used when combining load cases.

Following dialog shows the available options:



New

This button sets scaling factors for all load cases to 0.

All

All scalar fields with identical type as the first field are assigned the same given scaling factor.

Range

Within the given range of load cases all the load cases are assigned the same given scaling factor, if these scalar fields have the same type.

Load ID

The ID of a single load case must be given, which is assigned the given scaling factor.

Factor

The scaling factor to be assigned must be given.

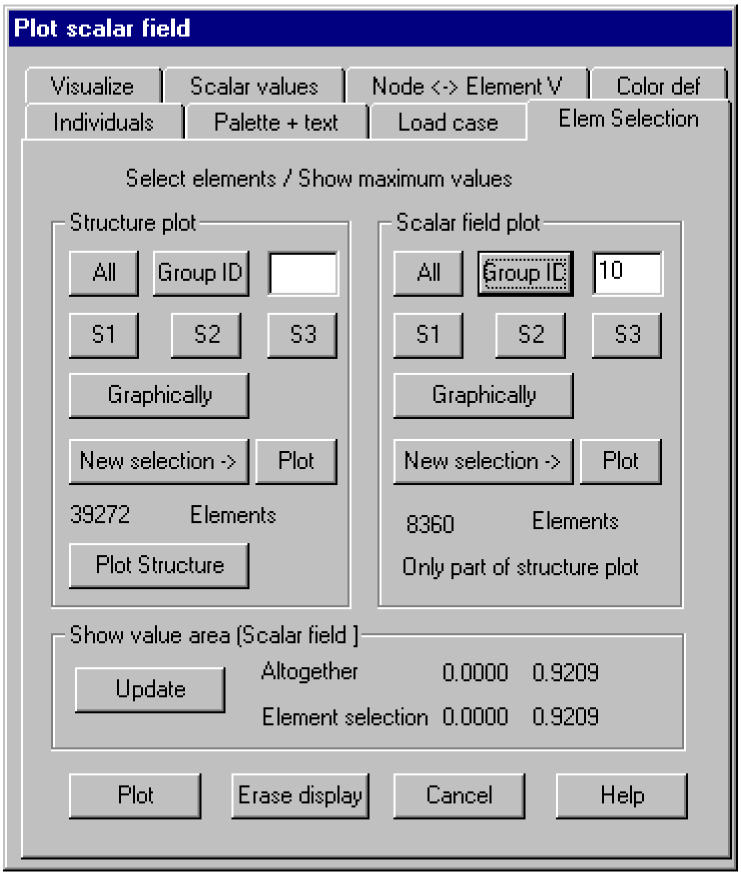
Plot

This button finally assigns the scaling factor to selected load cases and plots the distribution of scalar values.

Element selection: Select elements for structure and scalar field plot.

There is a difference between the element selection for structural plots and for scalar field plots. Only these elements are plotted which are contained within the current element selection for structure plots, but scalar fields are plotted only for those elements which are also contained within the element selection for the scalar field plot, remaining elements are plotted with colors assigned to the elements respectively a given color.

Following dialog shows the available options:



Structure plot

The element selection for plot of the structure has to be given. Pressing button “Plot structure” the current plot is updated.

Scalar field

The selection of elements for which scalar values should be plotted must be given. Only elements that are also in the selection for structure plot can be selected.

Show value area

Pressing button “Update”, the smallest and the largest value of the active scalar field are calculated for the elements in the selections and displayed.

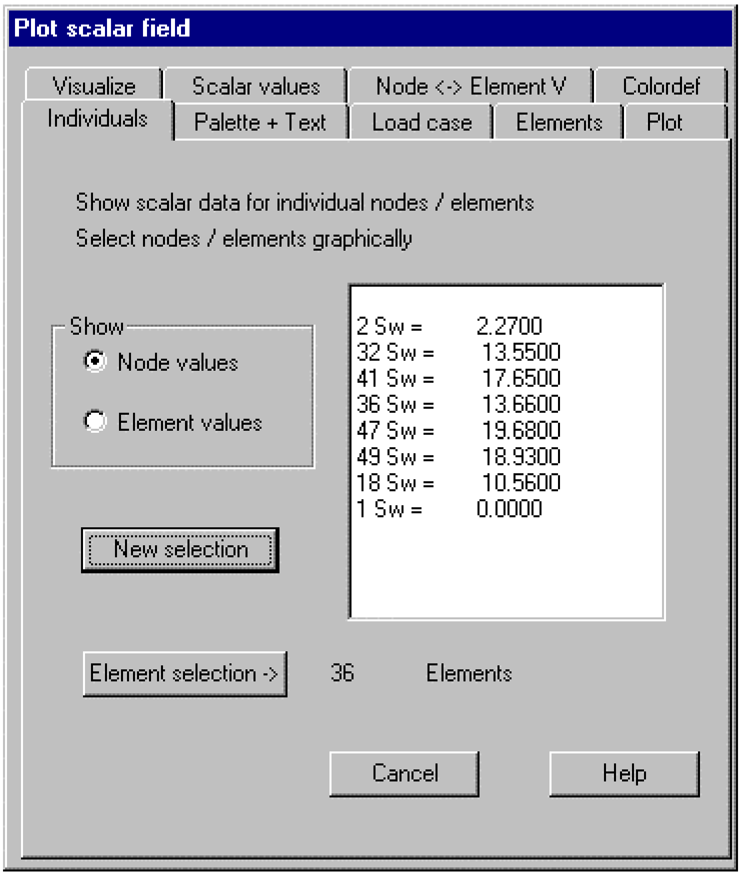
Update

Pressing this button will update the shown smallest and largest values.

Individuals: Show scalar data for individual nodes respectively elements

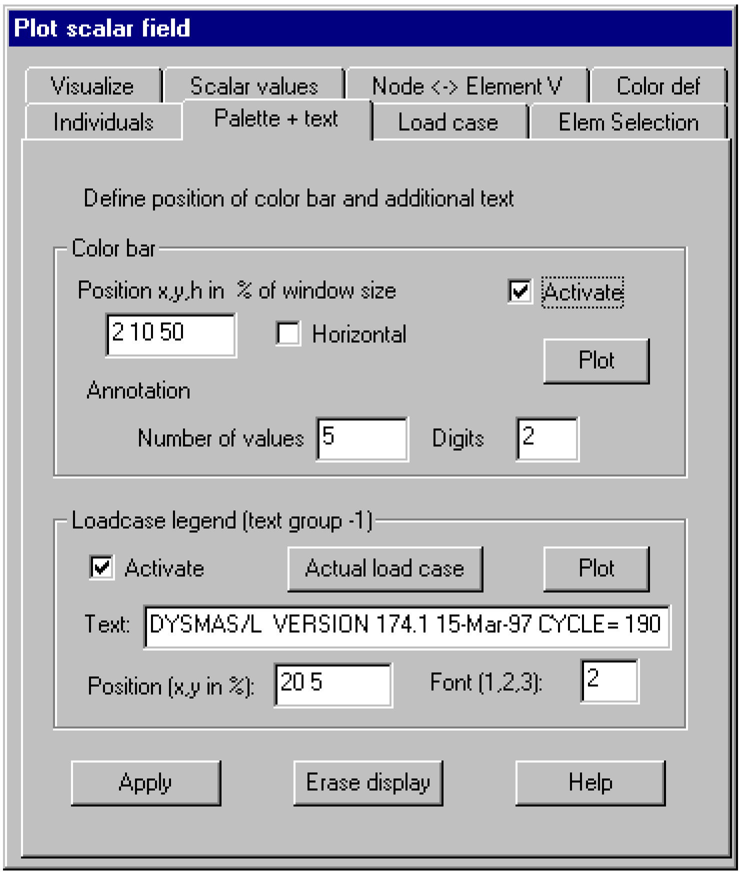
This property page allows the inquiry of scalar data on individual nodes or elements. At first, it must be selected whether node or element data should be shown. After clicking „New selection“ nodes respectively elements can be selected graphically by cursor and relevant data is shown within the list. All selectable nodes respectively elements are marked by a symbol. By pressing „Element selection“ the number of selectable nodes or elements can be reduced.

Following dialog shows the available options:



Color bar: Define position of color bar and define additional text

Following dialog shows the available options:



Color bar

Position x, y, h in % of window size: Position and height of the color bar in % of the window size must be given

Activate: The color bar will be plotted within a new layer. With this option set this layer won't be erased when refreshing the plot.

Horizontal: Plots color bar as a horizontal bar, otherwise as a vertical bar.

Annotation: To annotate the color bar, the number of color boxes to be annotated and the number of decimal digits can be given.

Plot: Color bar will be plotted.

Load case legend

For displaying a legend together with a load case following values can be provided:

Text: A single line of text can be given. If button „Actual load case" is pressed the current load case description given in the second line of the input file is used.

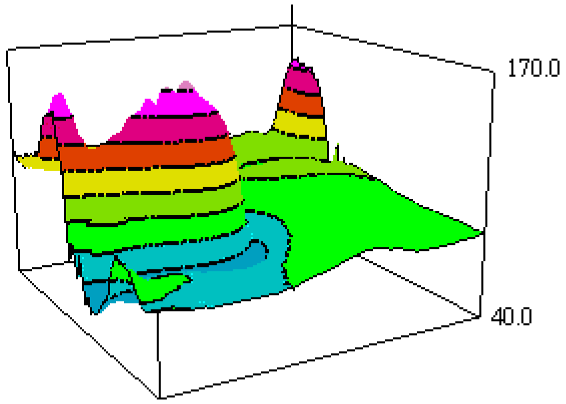
Position: The position of the text line must be given in percentage of window size.

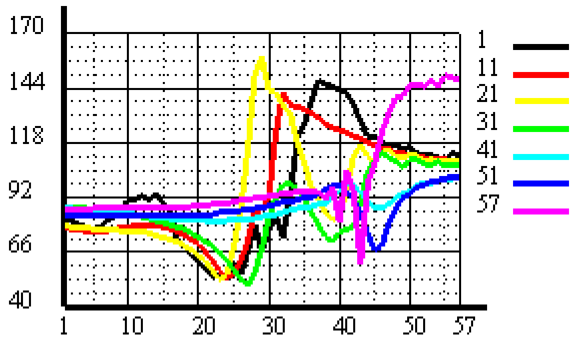
Font: The ID of the font to be used by the text has to be given.

The provided legend is saved as a text group -1 and plotted together with all active text groups (see command **Graphics text**).

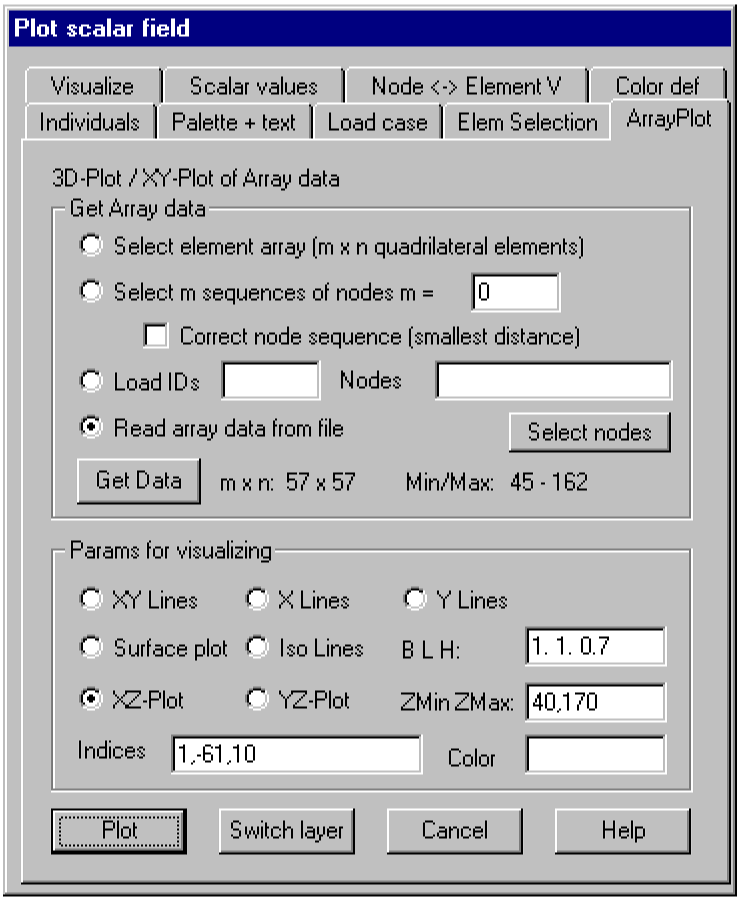
Array Plot: 3D-Plot or XY-Plot of Array Data

This command allows a 3D-Plot or a XY-Plot of scalar values that are given in a matrix of m lines and n columns (see image and demo "arrayview.dem").





Following dialog shows the available options:



Get Array Data

Select element array (m x n quadrilateral elements)

Using this option an area of m x n quadrilateral elements has to be selected in an element selection dialog. If body elements are contained in the selection, only those surfaces of them are used, of which the corner nodes are contained in the current node selection, that means, with body elements also a node selection must first be done and stored. The border of the selected element area is determined and graphically displayed. The 4 corner nodes of the selected 4 sided area are also marked by a symbol then the 2 corners of the edge, that is to be used as x axis in the plot must graphically be selected. The given scalar values at the corner nodes of the selected elements are stored in a matrix to be plotted.

Select m sequences of nodes

This option allows plotting the given scalar values at the nodes of 1 or more sequences of nodes (lines) in one XY Plot. The sequences of nodes have to be selected graphically, where each sequence has the same number (n) of nodes. The number (m) of sequences must be given in the input field. The scalar values in the selected nodes are then stored as array data in a (m/n) matrix, where each node sequence defines one line of the matrix. The selected nodes are ordered in the sequence they are selected, when selected by a single point selection, if selected by giving a rectangle or polygon area, they are ordered in the sequence of their internal node ID. If the option „Correct node sequence“ is marked, the nodes are newly ordered so that the distance of pursuing nodes is smallest. The lines are immediately plotted after selection.

Load IDs

This option can be used, if several load cases of node or element related scalar values are given. In the input field, the smallest and the largest ID of load cases to be used must be given. Additionally the nodes (elements) for which a XY-Plot of the data should be done have to be given in the input field (internal IDs). A continuous area may be given in the form k1,-k2,kd where k1 is the smallest, k2 the largest ID and kd the increment. Clicking button “Select graphically”, the nodes (elements) can be selected graphically. For each node (element) a matrix line is provided, that contains the scalar values of the node (element) given for the sequence of load cases. The number of array lines corresponds to the number of selected nodes (elements) and the length of the lines corresponds to the number of given load cases. This option is useful when the load cases correspond to time steps of an incremental calculation (data types 8 or 9), and the time dependent curves for several nodes (elements) should be plotted in a XY-Plot.

Read array data from a file

Clicking button "Read new file", a file with array data must be selected in a file selection dialog. In the first line of the file the number of lines (m) and columns (n) of the array must be given. Then m x n data values must follow in the sequence of lines (see demo file "arrayview.dat")

Get Data

Clicking this button, the array data is provided and stored in a matrix, corresponding to the selected option. The number of lines and columns (m x n) is shown in the dialog, also the smallest and largest value of the matrix. The data remains available until the dialog window for the plot of scalar values is closed.

Params for visualizing

XY Lines: With this option net lines are plotted in a 3D-Plot for all lines and columns of the matrix.

X Lines: With this option only net lines for the lines of the matrix are plotted, where all or individual lines may be selected.

Y L ines: With this option only net lines for the columns of the matrix are plotted where all or individual columns may be selected.

Surface plot: With this option, the surfaces between the lines are filled with a constant color. The index of the color must be given in the input field for colors.

Iso lines: With this option, iso lines are plotted in a 3D-Plot as shown in the above image. The number and the colors of the iso lines must be given in the dialog „Scalar values" respectively „Color definition". The kind of visualization can be altered in the dialog „Visualize" where following options can be given: Iso surfaces, Iso lines only, Angle of edges.

XZ-Plot: With this option an XY-Plot for selected lines of the matrix is done, as shown in the above image.

YZ-Plot: With this option a XY-Plot for selected columns of the matrix is done.

B L H: In the input field values for width, lengths and height of the 3D-Plot may be given. With XY-Plot, only values B and H are used.

ZMin ZMax: In the input field the smallest and the largest value to be used in z direction may be given.

Indices: For options „X lines", „Y lines", „XZ-Plot" and "YZ-Plot" indices of individual lines respectively columns of the matrix that should be plotted may be given. A continuous sequence of indices may be given in the form i1,-i2,id where i1 is the smallest, i2 the largest index and id is an increment. For example 1,-61,10 means each 10th index from 1 to 61. If the input field is empty, all lines respectively columns are plotted.

Color: In the input field color indices for the different curves may be given. If the field is empty, continuous indices are used, beginning by 1. For indices –1,-2,-3,... different line types are used.

Plot

Clicking this button a new graphics is done.

Switch Layer

Clicking this button, it can be switched between Array Plot and Structure plot, where only the corresponding OpenGL layers are switched on and off. This is done automatically, if a different property page of the dialog is used. If the dialog window is closed, the current array data is deleted and the used layer for the array plot is erased.

Plot vector fields

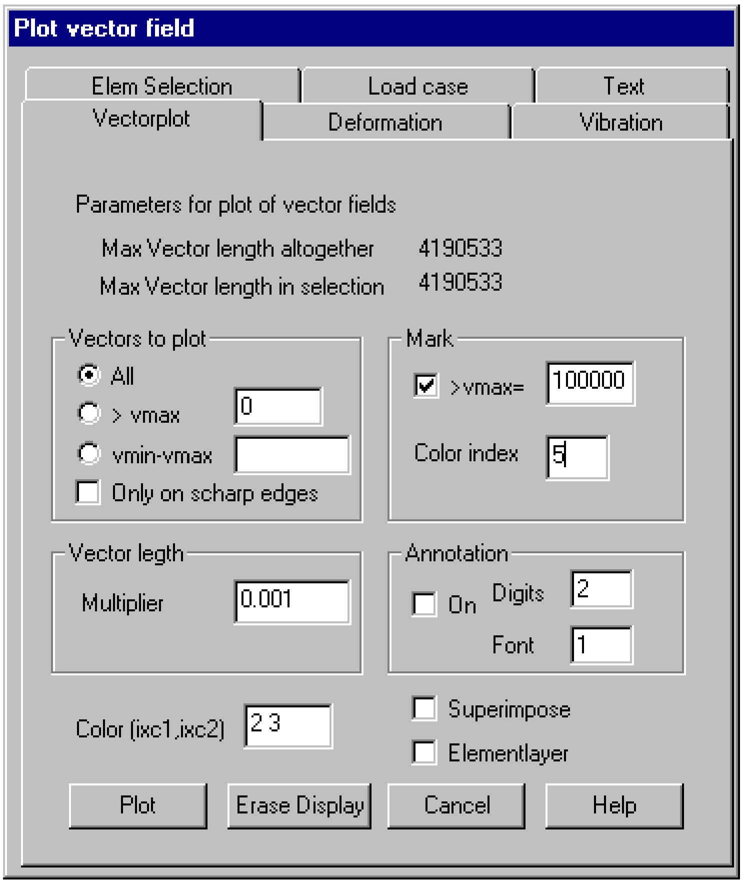
After giving the command **Vector field** a property dialog is popped up. This dialog remains active until it's closed by „Cancel".

Vector plot: Set parameters for following plots of vector fields

This property page sets parameters for plotting vector fields. Vectors are plotted as straight colored lines without arrows.

If the structure is displayed as a surface plot, only vectors belonging to visible nodes are plotted, however because of rounding errors the inquiry of visible nodes may not always be correct. If only sharp edges are plotted, alternatively the vectors in all nodes or only vectors on nodes of sharp edges may be plotted.

Following dialog shows the available options:



Vectors to plot

All: All vectors of the active vector field are plotted for selected elements.

> max: Only those vectors are plotted whose length is greater than the given value.

vmin - vmax: Only those vectors are plotted whose length is within the specified range. The selection of vector length makes it possible to do several overlapping plots with colors appropriate to vector length.

Only on sharp edges: If this option is set, only vectors in nodes belonging to sharp edges are plotted.

Vector length

The plotted length of the largest defined vector must be given. Vectors are multiplied by the given factor.

Mark

This options makes it possible to emphasize vectors whose length exceeds a given value. The length and color index must be given.

Annotation

Optionally vectors can be annotated by the value of their length. The number of digits after decimal point and the font ID must be given.

Color

The color index used for plotting vectors must be given. In case of vector crosses 2 indices must be provided.

Superimpose

With this option set the plots are done within distinct layers. By this way vectors for different load cases can be plotted simultaneously using different colors.

Element layer

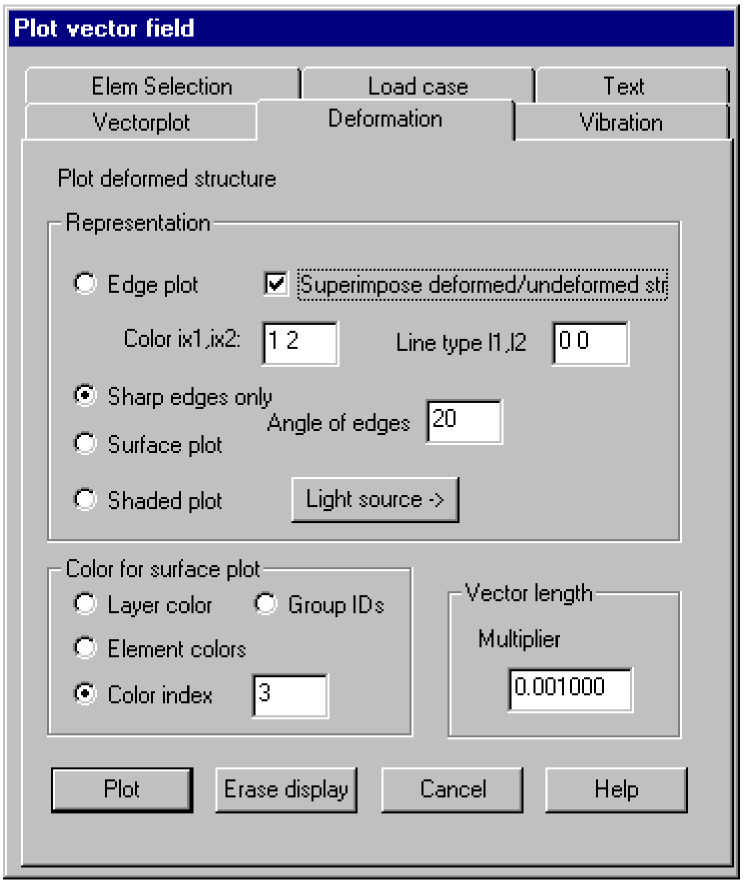
If this option is set and there are given layer IDs for different parts of the structure, the vectors are plotted in different layers, so that the vectors of different part of the structure may easily be shown or hidden.

The plot is done by pressing „Plot“. Button „Erase Display“ erases all display lists.

Deformation: Plot deformed structure

Vectors associated with nodes can be interpreted as translation vectors for these nodes. With this property page these translations can be visualized as structural deformations.

Following dialog shows the available options:



Representation

Edge plot: The structure is plotted as a wire frame. If option „Superimpose deformed / non-deformed structure" is set, the non-deformed structure is plotted with color index ix1 and line type l1 simultaneously with the deformed structure with color index ix2 and line type l2.

Sharp edges only: Only those model edges are plotted whose angle between the normal vectors of adjacent element surfaces exceeds the given value.

Surface plot: Plot deformed structure as a surface plot.

Shaded plot: Plot deformed structure as a shaded plot. By clicking button „Light source" a dialog is popped up to define additional parameters for the light source.

Angle of edges: An angle must be given for the option "Sharp edges only" and may be given for the options "Surface plot" and "Shaded plot".

Color selection for surface plots

For surface plots the colors to be used have to be selected.

Vector length

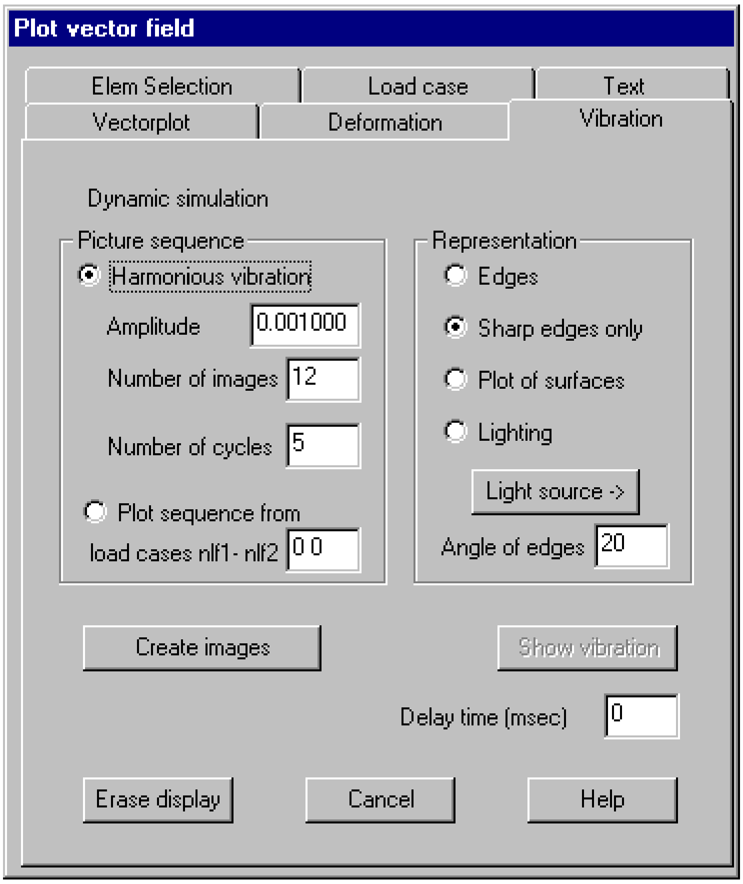
The length that will be used for plotting the largest vector must be specified. The vectors are multiplied by the given value.

Button „Plot" plots the deformed structure; „Erase display" erases all display lists.

Vibration: Dynamical simulation

Nodal displacements can be interpreted as amplitudes of a harmonious vibration.

Following dialog shows the available options:



Harmoniously vibration

Amplitude: The plotted length for the largest vector must be given; vectors are multiplied by this factor.

Number of images: The number of images for half the vibration cycle must be given.

Number of cycles: The number of vibration cycles to be plotted must be given.

Plot sequence from

A non harmoniously vibration can be simulated by a sequence of images. The images are created as node displacement plots defined by vector fields given in continuous load cases (for example type of input file 11). The first and last load case ID must be given.

Representation

The parameters are the same as with the command **Deformation**.

Create images

Clicking this button creates the sequence of images.

Show vibration

Clicking this button shows previously created images as an animation. The refresh rate heavily depends on the size of the window and the installed graphics adapter. In case of a too quick display a delay time between images (ms) can be given.

Elements: Element selection for structure respectively vector field plot

This property page allows selections for structure and vector field plot, so it's possible to plot only vectors for selected elements. The dialog is the same as with plot of scalar fields.

Load case: Define load case combinations

Vector fields of the same type can be combined with a new load case. This page allows selecting the load case to be plotted and to define the scale factors to be used when combining several load cases. The dialog is the same as with plot of scalar fields.

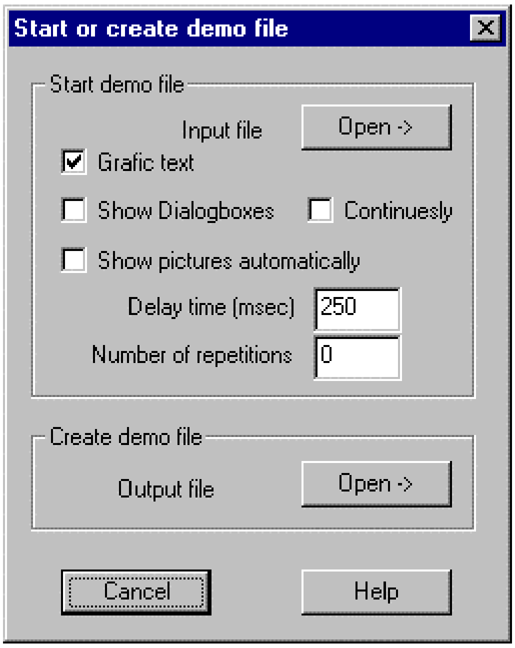
Demo-files, tutorial

Command Demo File

A sequence of commands of a session can be recorded in a file for later reproduction as a demonstration. **Demo files** automatically get the extension .dem. Optionally as a tutorial, also the dialog boxes associated with the commands can be shown with the used parameter values in a demonstration.

Demonstrations are started with the command **Demo File** in menu group **Help.**

Following figure shows the available options:



Start demo file

Graphic text: Associated with each command is a line of text, which shows the command name and its function. If this option is active, the text is displayed in the graphics window

Show dialog: With this option active, the dialog associated with the command is popped up to show the command parameters.

Continuously: With this option not set, each different dialog box is only shown once, with this option set the dialog box is shown again.

Show pictures automatically: With this option set, the next command is executed automatically. A dialog is shown, where you can stop the automatic sequence. With this option not set, the command sequence is stopped after each new graphics output. In a dialog you confirm continuation or stop the run or switch to automatic sequence.

Delay time: The time for delay after each new graphics output must be given in milliseconds.

Number of repetition: If a number m > 0 is given, the demonstration is automatically restarted m times after the end of the file is reached.

Open: After setting the options, the run is started with this button. The file name has to be selected in a file selection box.

Create demo file

After clicking „Open“, the name of the output file has to be given in a file selection box. Following commands are recorded in this file, until the file is closed with the same button, which is now labeled „Close“.

Note: Recording of commands is not implemented for all possible actions. A demo file cannot be used as a restart file.

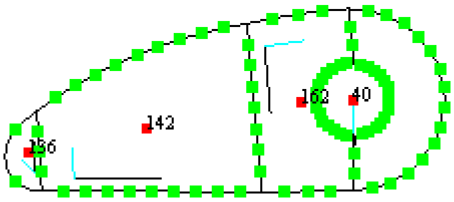
Tutorial

Folder \demo contains files that demonstrate the usage of some commands and can be used as tutorials. The files are started with command „Demo File“. If a connection of files with extension .dem with the program „makrosa.exe“ is registered in Windows, a demonstration can be started by double clicking a demo file.

Following some of the available demo files are described.

Demofile: subdivision.dem

This example shows the subdivision of a structure with 4 macro elements into a FE model. Subdivision parameters are assigned with the command „Pattern“.



Demofile: Intersection.dem

The macro and FE models show in the following figure are generated only by using MAKROS commands.

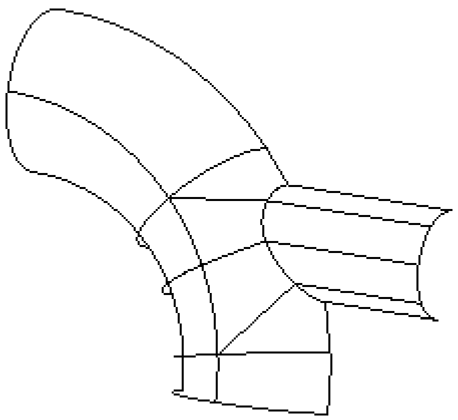
The generation is done in following steps:

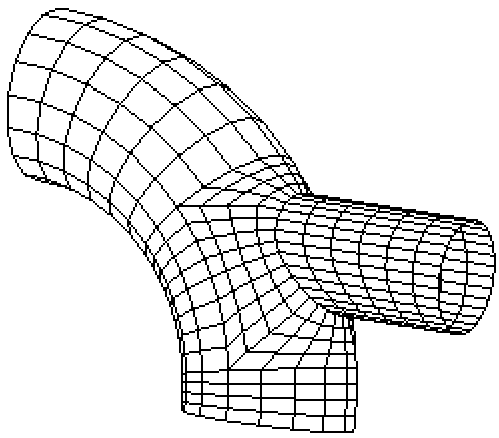
Command and Function:

Coordinate System	Define local toroidal coordinate system
Define Nodes	2 individual nodes in toroidal coordinates
Define Nodes	Divide line between this nodes in toroidas coordinates
3D Ext/Translation	Create additional nodes on toroid by translation
Define Elements	Define macro elements graphically
Delete	Delete point elements
Coordinate System	Define local cylindrical coordinate system
Define Nodes	2 individual nodes in cylindrical coordinates
Define Nodes	Divide line between this nodes in cylindrical coordinates
3D Ext/Translation	Create additional nodes on cylinder by translation

MAKROS-A

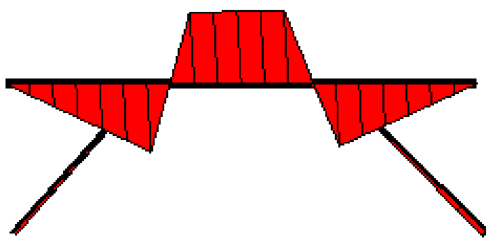
Intersection	Move nodes on cylinder on to torus surface
Define Elements	Define elements graphically
Delete	Delete point elements
Define Nodes	Create intermediate nodes by division of lines in toroidal coordinates
Delete	Delete 2 macro elements
Define Elements	Define elements graphically
Delete	Delete point elements
Pattern	Assign subdivision parameters
Subdivision	Create FE model by dividing macro elements
Smooth	Smooth nodes on torus surface
Mirroring	Mirror elements in plan Z = 0

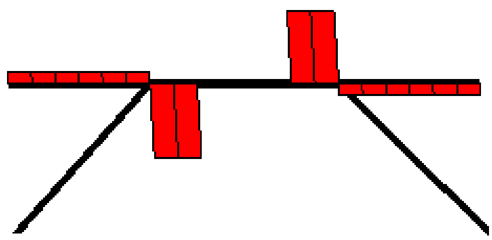




Demofile: nastran1.dem

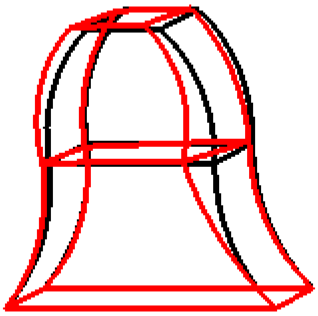
This example shows pre and post processing of the following frame structure.



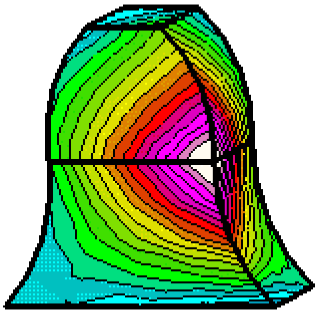


Demofile: nastran2.dem

This example shows pre and post processing of the following solid model.



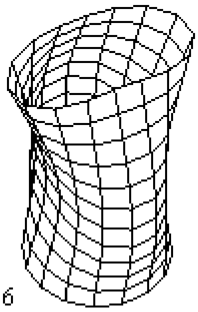
DEFORMATION



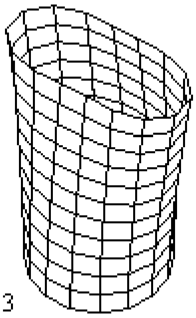
VON MISES STRESSES

Demofile: nastran4.dem

This example shows pre and post processing of the following shell model.



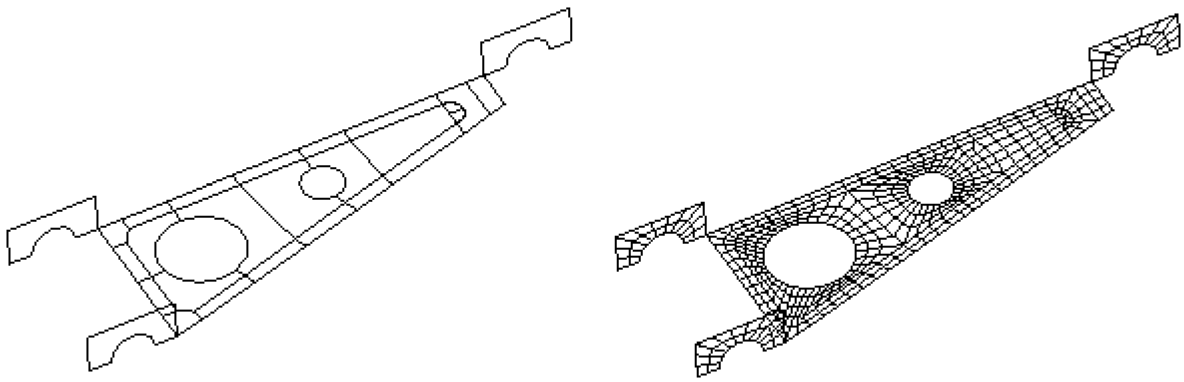
Mode 6



Mode 3

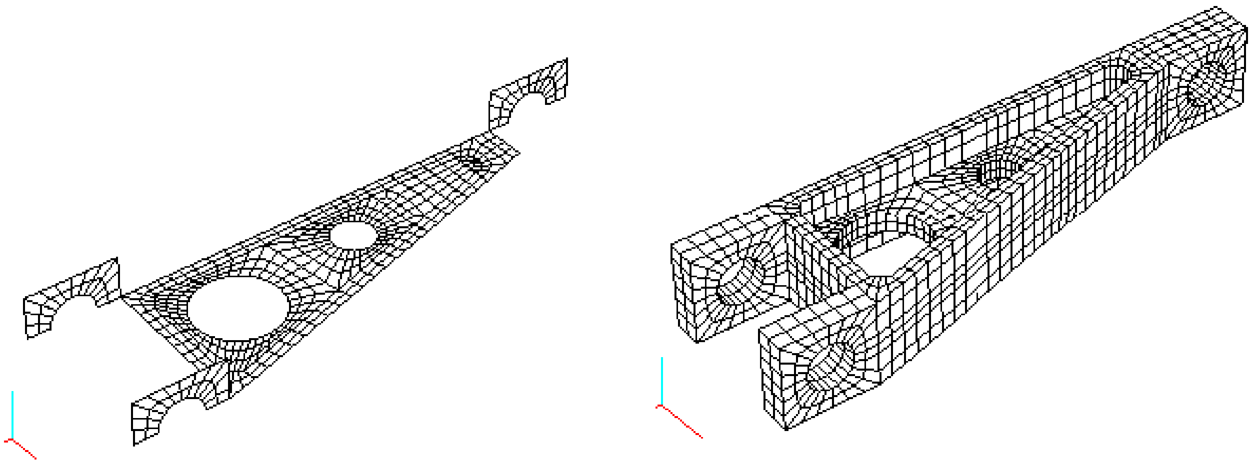
Demo file: subdivision.dem

The macro model shown in the following figure is read from a DXF-File. With command „**Pattern**“ subdivision parameters for a regular subdivision of the macro elements are assigned to get the shown structure.



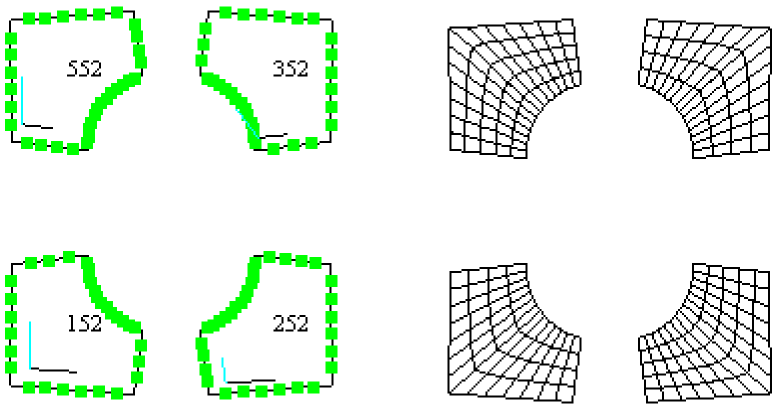
Demo file: 3Dextension.dem

This example shows, how to create a solid model out of a surface model. Used are the commands **3DExtension/Translation** and **Mirroring** to generate the shown FE model.



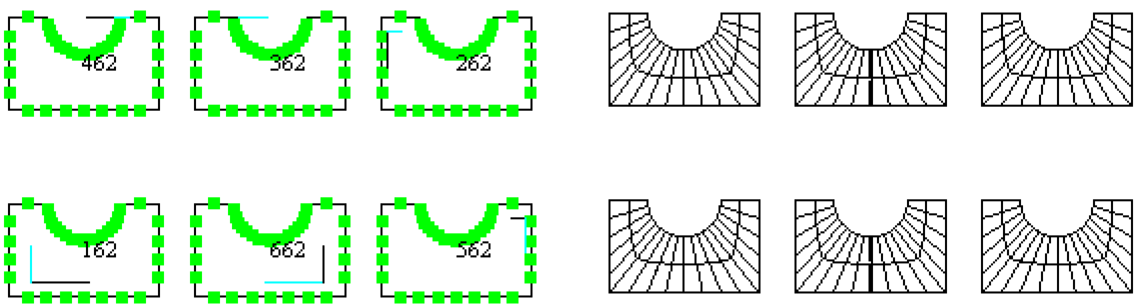
Demo file: patern15x.dem

This example shows, which pattern id to use to get a regularly subdivision of macro elements with 5 vertices.



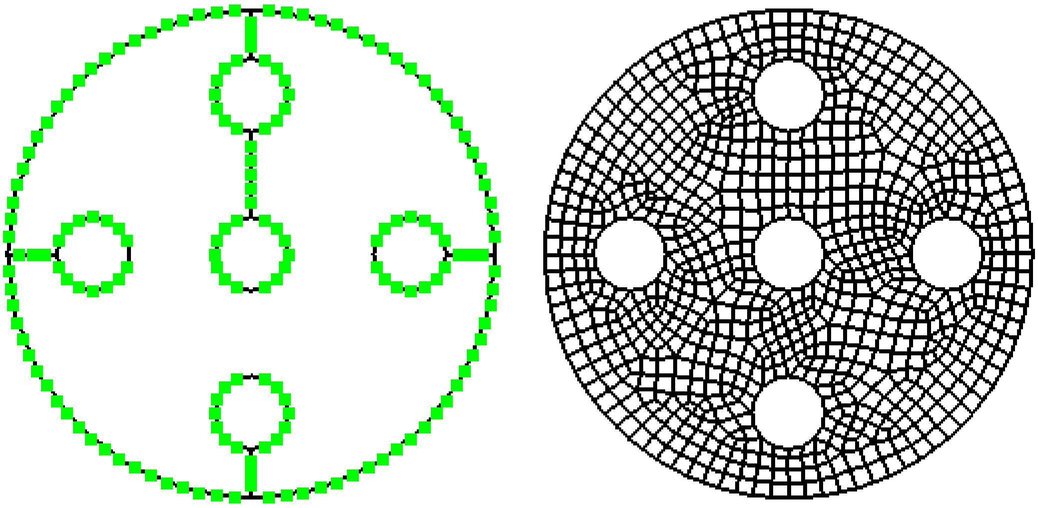
Demo file: patern16x.dem

This example shows, which pattern id to use to get a regularly subdivision of macro elements with 6 vertices.



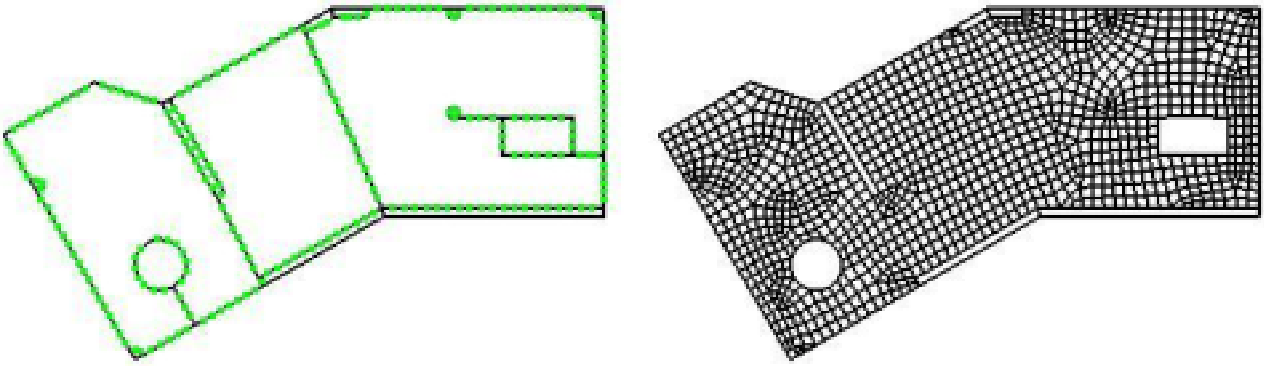
Demofile: typ400_1.dem, typ400_2.dem, typ400_3.dem

Element type 400: free mesh generation of plane surfaces.



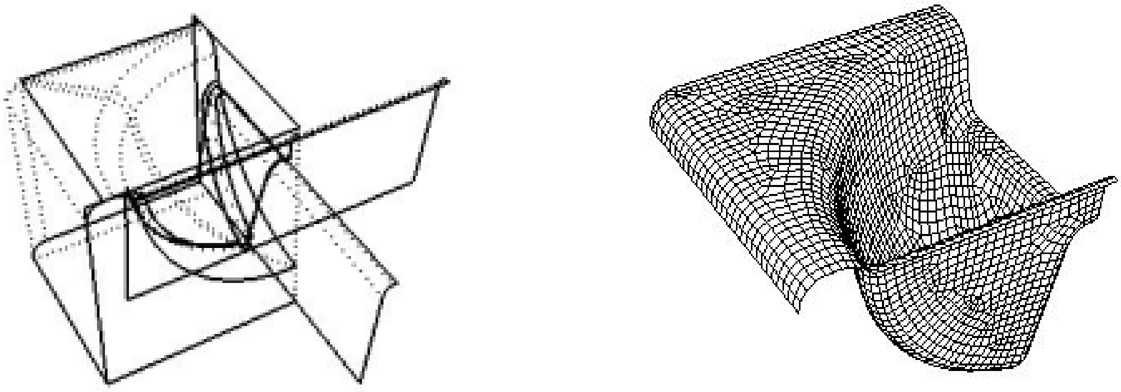
Demofile: dplatte.dem

Element type 400: free mesh generation of plane surfaces.



Demo file: vdafs.dem

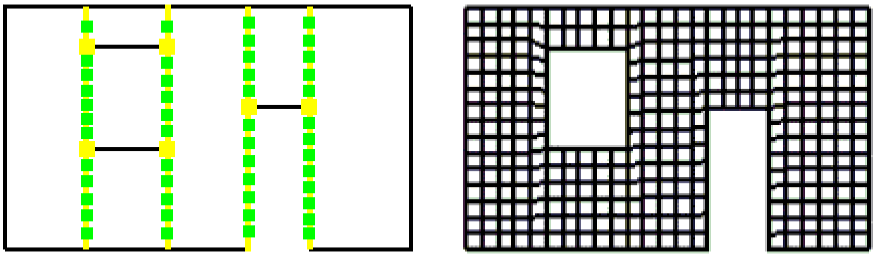
Generation of a macro and finite element model using a CAD model in VDAFS format.



Demofile: referencepoint.dem

Subdivision of macro elements using reference points.

Order of specification: Define position of reference points, subdivide all edges with command „Division“, control the finite element net with command „Subdivision“ and option test some elements, correct the number of intermediate nodes for some parts of the edges with reference points using command „reference points“.

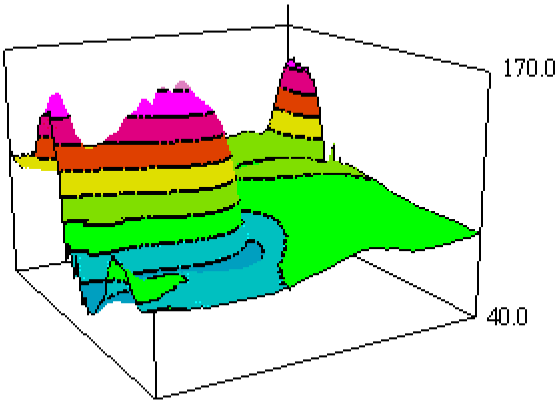


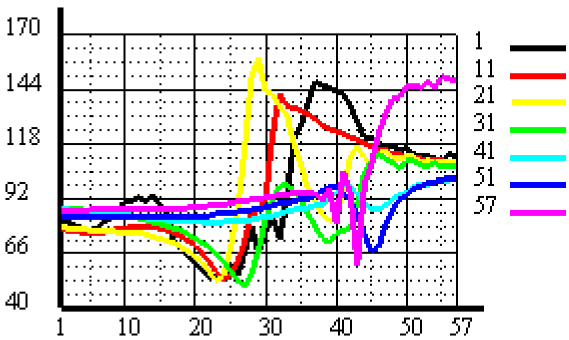
Demofile: nastran3.dem

Pre- and post processing of a plate.

Demofile: arrayplot.dem

3D-Plot und XY-Plot of array data (see command „Array Plot“)





Demofile: defkoor.dem

Definition of boundary coordinate systems.

Demofile: netz.dem

Local net refinement

Demofile: mak01903.dem

Post processing of a big structure

Demofile: pattern150.dem

Subdivision of macro type 150.

Demofile: plate.dem, typ400_1.dem, typ400_2.dem, typ400_3.dem

Subdivision of macro type 400.

Demofile: post1.dem, post2.dem, arrayview.dem

Post processing.

Demofile: translation.dem, move.dem

command **3DExtension/Translation**.

Demofile: demo.dem

Combination of multiple demo files.

Appendix A: Linux / Unix

Makros is available with identical functionality also for PCs with LINUX operating system and for some Unix workstations, only the dialogs have been adapted to X-WINDOW.

A.1 Dialogs

For the dialogs Motif is used. The dialogs are to a large extent identical with the shown Windows dialogs in the manual. An exception are drop down lists, here mostly only an editable text window is given, where the required value for the option must be typed in (examples are type ID of elements or coordinate ID of a local coordinate system).

A.2 Resource File

The appearance of the dialogs can in some degrees be altered with some variables, that must be given in a file „**Makrosa.res**“. The location of the file must be the same as the execution file of makrosa.

The Resource file has the following structure:

Lines beginning with the character # contain only comments.

Definition of variables are given in one line with first the name of the variable followed by the value of the variable, optionally followed by comment beginning with the character #.

Following variables are used:

Variables that define the background color of some dialog elements:

Color values may be given as names as defined in the file „/usr/lib/X11/rgb.txt“ or as rgb values in the form #RRGGBB, for example color „gray95“ has the rgb value #F2F2F2. Following variables for background colors are used:

bgDlgBox: background color of the dialog box,

bgButtonM: background color of menu buttons,

bgButtonH: background color of the button of the active dialog by overlying dialogs in one window,

bgButton: background color of dialog buttons,

bgText: background color of input lines,

Variables that determine the used font with some dialog elements:

Using the X command „xlsfonts“ the names of all installed fonts may be listed and using the command „xfd –fn font name“ the appearance of some fonts may be shown. Following variables for fonts are used:

fontButtonM: font of menu buttons,

fontButton: font of dialog buttons,

fontToggle: font of toggle buttons,

fontLabel: font of labels,

fontLabelBig : font of emphasized labels,

grFont1: font 1 of graphical text with OPENGL,

grFont2:	font 2 of graphical text with OPENGL,
grFont3:	font 3 of graphical text with OPENGL.

Variable that defines the path of the html help files:

Provided that the html help files are not located in subdirectory „htmlhelp“ of makrosa the full path of these files must be given using the variable „helpPath“ .

Default values of the Resources:

For the resources the following default values are used if the file „makosa.res“ is not found or no values for the variables are given:

bgButtonM	gray87
bgButton	gray87
bgText	white
bgButtonH	gray80
bgDlgBox	gray87

fontButtonM	*-helvetica-bold-r-normal--12-120-75-75-*
fontButton	*-helvetica-bold-r-normal--12-120-75-75-*
fontToggle	*-helvetica-bold-r-normal--12-120-75-75-*
fontLabel	*-helvetica-medium-r-normal--12-120-75-75-*
fontLabelBig	*-helvetica-bold-r-normal--14-140-75-75-*
grFont1	7x14
grFont2	9x15
grFont3	9x15bold

A.3 Showing help information

Clicking help button in the dialog Netscape is invoked with the title of the corresponding html file as argument. First invocation of Netscape requires some time, do not terminate Netscape after first invocation, only iconify if, because if Netscape is invoked again, the argument “-remote” is used, that requires that Netscape is still running.

The content of the html help files are identical with the corresponding chapters of this manual. The files are generated out of the doc files using Word for Windows. Because of this, some formats are not correctly shown by Netscape.

Appendix B: Keywords (Links)

General

Elementtypes

Definition of macro elements within AutoCAD

Subdivision patterns

MAKROS-A commands overview

Commands processing the entire structure

Commands to generate new nodes and elements

Handling CAD data in VDAFS format

Subdivision of macro elements into finite elements

General commands

Commands for graphical representation of the structure

Commands specifying loads, constraints, element properties and materials

Interfaces to FE-Programs

Postprocessing

Appendix A