

# **OPERA-3D REFERENCE MANUAL**

Vector Fields Limited  
24 Bankside  
Kidlington  
Oxford OX5 1JE  
England

Copyright © 1999-2002 by Vector Fields Limited, England

This document was prepared using Adobe® FrameMaker®.

UNIX is a trademark of X/Open Company Ltd.

HPGL is a trademark of Hewlett-Packard Incorporated

PostScript is a trademark of Adobe Systems Incorporated

Windows 95/98/NT is a trademark of Microsoft Corporation

X Window System is a trademark of Massachusetts Institute of Technology

I-DEAS is a trademark of Structural Dynamics Research Corporation

OpenGL is a trademark of Silicon Graphics Corporation

All other brand or product names are trademarks or registered trademarks  
of their respective companies or organisations.

# CONTENTS

## *Chapter 1* *System Overview*

Introduction .....	1-1
Supported Analysis Programs .....	1-2
Program Limits .....	1-3

## *Chapter 2* *User Interface*

Introduction .....	2-1
Keyboard Input .....	2-2
Output Files .....	2-3
Commands and Parameters .....	2-4
The Help Character (!) .....	2-5
Parameter Assignment .....	2-6
Parameter Values .....	2-8
Command Interpreter Errors .....	2-13
Confidence Level .....	2-14
Prompted free format Input .....	2-16
Built-in Commands .....	2-17
Command Separator and Comments .....	2-30
Euler Angles .....	2-31

*Chapter 3*  
*OPERA-3d Geometric Modeller*

Introduction .....	3-1
Geometric Modelling .....	3-3
Summary of Commands .....	3-15
The <b>ANALYSISDATA</b> Command .....	3-18
The <b>ARC</b> Command .....	3-22
The <b>BEDSTEAD</b> Command .....	3-24
The <b>BLEND</b> Command .....	3-26
The <b>BLOCK</b> Command .....	3-27
The <b>BOUNDARY</b> Command .....	3-28
The <b>BRICK8</b> Command .....	3-31
The <b>BRICK20</b> Command .....	3-33
The <b>CELLDATA</b> Command .....	3-35
The <b>CHECK</b> Command .....	3-37
The <b>CLEAR</b> Command .....	3-38
The <b>COLOUR</b> Command .....	3-39
The <b>COMBINE</b> Command .....	3-40
The <b>COMMENT</b> Command .....	3-42
The <b>CONDUCTOR</b> Command .....	3-43
The <b>CONTOUR</b> Command .....	3-45
The <b>CYLINDER</b> Command .....	3-47
The <b>DBCASEDATA</b> Command .....	3-48
The <b>DELETE</b> Command .....	3-49
The <b>DRIVE</b> Command .....	3-50
The <b>EDGEDATA</b> Command .....	3-53
The <b>END</b> Command .....	3-54
The <b>EXPORT</b> Command .....	3-55
The <b>FACEDATA</b> Command .....	3-56
The <b>FILL</b> Command .....	3-58
The <b>FILTER</b> Command .....	3-59
The <b>FITTEDCPE</b> Command .....	3-61
The <b>HELICALEND</b> Command .....	3-63
The <b>HIDE</b> Command .....	3-65
The <b>HISTORY</b> Command .....	3-66
The <b>IMPORT</b> Command .....	3-67

The <b>LABEL</b> Command .....	3-68
The <b>LCS</b> Command .....	3-69
The <b>LIST</b> Command .....	3-70
The <b>LOAD</b> Command .....	3-71
The <b>MATERIALS</b> Command .....	3-73
The <b>MESH</b> Command .....	3-77
The <b>MODEL</b> Command .....	3-79
The <b>MOUSE</b> Command .....	3-81
The <b>PERIODICITY</b> Command .....	3-82
The <b>PICK</b> Command .....	3-85
The <b>PICTURE</b> Command .....	3-88
The <b>PRECISIONDATA</b> Command .....	3-89
The <b>PRINT</b> Command .....	3-90
The <b>PRISM</b> Command .....	3-91
The <b>RACETRACK</b> Command .....	3-92
The <b>REDO</b> Command .....	3-94
The <b>RENAME</b> Command .....	3-95
The <b>SAVE</b> Command .....	3-96
The <b>SELECT</b> Command .....	3-98
The <b>SOLENOID</b> Command .....	3-103
The <b>SOLVERS</b> Command .....	3-105
The <b>SPHERE</b> Command .....	3-107
The <b>STRAIGHT</b> Command .....	3-108
The <b>SWEEP</b> Command .....	3-110
The <b>TANGENTIALCPE</b> Command .....	3-112
The <b>THREED</b> Command .....	3-114
The <b>TITLE</b> Command .....	3-116
The <b>TORUS</b> Command .....	3-117
The <b>TRANSFORM</b> Command .....	3-118
The <b>UNDO</b> Command .....	3-121
The <b>VARIABLE</b> Command .....	3-122
The <b>VECTOR</b> Command .....	3-123
The <b>VERTEXDATA</b> Command .....	3-125
The <b>VOLUME</b> Command .....	3-126
The <b>WCS</b> Command .....	3-128
The <b>WINDOW</b> Command .....	3-129

*Chapter 4*  
*OPERA-3d Pre Processor*

Introduction .....	4-1
The Graphical User Interface .....	4-3
Pre Processor Quick Reference Guide .....	4-10
The <b>BHDATA</b> Command .....	4-15
The <b>CHECK</b> Command .....	4-18
The <b>CLEAR</b> Command .....	4-21
The <b>COLOUR</b> Command .....	4-22
The <b>CONDUCTOR</b> Command .....	4-24
The <b>DEFINE</b> Command .....	4-49
The <b>DEVICE</b> Command .....	4-95
The <b>DISPLAY</b> Command .....	4-97
The <b>DUMP</b> Command .....	4-104
The <b>EDIT</b> Command .....	4-109
The <b>END</b> Command .....	4-112
The <b>EXTEND</b> Command .....	4-113
The <b>FILL</b> Command .....	4-115
The <b>HELP</b> Command .....	4-117
The <b>IDEAS</b> Command .....	4-118
The <b>LABEL</b> Command .....	4-125
The <b>MATERIALS</b> Command .....	4-129
The <b>MESH</b> Command .....	4-133
The <b>MODIFY</b> Command .....	4-135
The <b>READ</b> Command .....	4-149
The <b>REDEFINE</b> Command .....	4-151
The <b>SLIP</b> Command .....	4-152
The <b>SOLVERS</b> Command .....	4-153
The <b>TABLE</b> Command .....	4-170
The <b>THREED</b> Command .....	4-171
The <b>TITLE</b> Command .....	4-176
The <b>TRANSFORM</b> Command .....	4-178
The <b>WRITE</b> Command .....	4-179

## Chapter 5

### *OPERA-3d Post Processor*

Introduction .....	5-1
Post Processor Quick Reference Guide .....	5-3
Field Component Evaluation .....	5-10
Post Processor Data Files .....	5-18
The <b>ACTIVATE</b> Command .....	5-21
The <b>ARC</b> Command .....	5-24
The <b>ARITHMETIC</b> Command .....	5-26
The <b>BHDATA</b> Command .....	5-27
The <b>BODY</b> Command .....	5-28
The <b>CARTESIAN</b> Command .....	5-31
The <b>CIRCLE</b> Command .....	5-33
The <b>CLEAR</b> Command .....	5-34
The <b>COLOUR</b> Command .....	5-35
The <b>COMBINE</b> Command .....	5-37
The <b>CONDUCTOR</b> Command .....	5-39
The <b>END</b> Command .....	5-45
The <b>ENERGY</b> Command .....	5-46
The <b>FIT</b> Command .....	5-50
The <b>GRAPH</b> Command .....	5-52
The <b>GRID</b> Command .....	5-55
The <b>IDEAS</b> Command .....	5-57
Integration Commands .....	5-60
The <b>INTEGRATE</b> Command .....	5-62
The <b>LINE</b> Command .....	5-65
The <b>LOAD</b> Command .....	5-67
The <b>MAP</b> Command .....	5-68
The <b>MOUSE</b> Command .....	5-71
The <b>PICTURE</b> Command .....	5-72
The <b>PLOT</b> Command .....	5-73
The <b>POINT</b> Command .....	5-76
The <b>POLAR</b> Command .....	5-77
The <b>PRINT</b> Command .....	5-79
The <b>SELECT</b> Command .....	5-80
The <b>SET</b> Command .....	5-85

The <b>SHOW</b> Command .....	5-88
The <b>SURFACE</b> Command .....	5-90
The <b>SYSVARIABLE</b> Command .....	5-92
The <b>TABLE</b> Command .....	5-97
The <b>TITLE</b> Command .....	5-100
The <b>THREED</b> Command .....	5-101
The <b>TRACK</b> Command .....	5-104
The <b>UNITS</b> Command .....	5-107
The <b>VIEW</b> Command .....	5-110
The <b>VOLUME</b> Command .....	5-116
The <b>WINDOW</b> Command .....	5-119

# *Chapter 1*

## **System Overview**

### **Introduction**

---

---

OPERA-3d (an **OP**erating environment for **Electromagnetic Research and Analysis**) is the pre and post processing system for well known electromagnetics analysis programs including TOSCA, ELEKTRA, SCALA, CARMEN and SOPRANO.

Finite element discretisation forms the basis of the methods used in these analysis programs. This widely applicable technique for the solution of partial differential equations requires special enhancements to make it applicable to electromagnetic field calculations. Access to these features is supported by the OPERA-3d Geometric Modeller and pre processor. These program provide facilities for the creation of finite element models, specification of complicated conductor geometry, definition of material characteristics including for example, non-linear and anisotropic descriptions and graphical displays for examination of the data.

Similarly, the OPERA-3d post processor provides facilities necessary for calculating electromagnetic fields. As well as displaying field quantities as graphs and contour maps, the OPERA-3d post processor can calculate and display many derived quantities and can plot particle trajectories through the calculated fields.

This Reference Manual describes in detail the OPERA-3d Modeller and the pre and post processors: chapter 2 describes the common User Interface; chapter 3 is a Reference Manual for the Modeller, chapter 4 for the pre processor and chapter 5 for the post processor.

The rest of this chapter gives summaries of the analysis programs which are supported. Full details of the analysis programs can be found in the User Guide.

## Supported Analysis Programs

---

There are currently 5 supported analysis packages: CARMEN, ELEKTRA, SOPRANO, SCALA and TOSCA.

### CARMEN

CARMEN analyses transient electromagnetic fields in rotating machines, with constant rotor speeds.

### ELEKTRA

ELEKTRA analyses time dependent electromagnetic fields, including the effects of eddy currents, in three dimensions. There are 3 analysis options: the time variation can be transient (TR), steady state ac (SS) or eddy currents can be induced in moving conductors with a specified linear or rotational velocity in the presence of a static field (VL).

### SCALA

SCALA analyses electrostatic fields taking into account the effects of space charge created by beams of charged particles.

### SOPRANO

SOPRANO analyses high frequency electromagnetic fields in three dimensions. There are two analysis options: steady state ac or eigenvalue extraction.

### TOSCA

TOSCA solves non-linear magnetostatic or electrostatic field and current flow problems in three dimensions. It has been in use for many years, but is being continually improved to increase its accuracy and efficiency. TOSCA uses a formulation based on total and reduced scalar potentials, solved using finite elements.

## Program Limits

---

The OPERA-3d pre processor has a limit on the maximum numbers of entities which can be created and the Standard versions of the analysis programs have a limit on the size of finite element mesh which can be analysed. The Professional versions of the analysis programs and post processor will “grow” to accommodate the data up to the limit of available virtual memory (swap space).

The current limits of the three sizes are given in the following table:

OPERA-3d Program Limits				
		Standard	Professional	
		Size 1	Size 2	Size 3
Pre processor internal database	entities <sup>a</sup>	200000	400000	2000000
Pre processor	conductors <sup>b</sup>	2000	2000	20000
CARMEN, ELEKTRA, SCALA, SOPRANO, TOSCA and post processor	nodes	50000	no limit	no limit

a. Entities = total number of points+lines+facets+volumes.

b. The limit on conductors only applies to the display in the pre processor.

- UNIX workstations are supplied with Size 2.



# Chapter 2

## User Interface

### Introduction

---

---

The interactive Geometric Modeller, pre and post processors of OPERA-3d have user interfaces which comprises both a command line, and a Graphical User Interface (GUI).

The GUI generates text commands which have the same syntax as the commands which can be typed in directly at the keyboard. There are some features which can be used only from the keyboard. This chapter gives full details of the command line interface. In subsequent chapters, the keyboard commands are described with indications, where appropriate, of the corresponding GUI interaction. The use of the GUI in the Modeller and post processor is described in the User Guides; the use of the GUI for the pre processor is given in “[The Graphical User Interface](#)” on page 4-3.

Within this manual, different fonts are used to differentiate between input and output of various types. The program’s commands, parameters and keywords are shown in **SANS-SERIF FONT**; **input** and output from the program in **bold** and normal weight teletype font. File names are shown in a *slanted font*. GUI items are shown in a **narrow tele-type font**.

## Keyboard Input

---

A typical keyboard input consists of a command to perform some action, together with parameters that determine how the action is performed. At other times the input is a list of 'free-format' keywords or numbers which provide additional input to an earlier command. Keyboard input is requested by a prompt of the form

```
name >
```

where name is the name of the program or program section being used or sometimes is a question to be answered. Alphabetic input can be in either upper or lower case.

Even in keyboard mode, some commands require graphical input. This is provided by positioning the cross-shaped cursor on the graphics window and typing a key on the keyboard or a mouse button.

Some commands are 'built-into' the command interpreter. Whenever a prompt of the form given above is issued, built-in commands can be typed, by starting the input line with \$.

## Output Files

---

All user input and the responses from the program are stored in dialogue files:

- *Opera3d\_Modeller\_n.lp* for the modeller
- *Opera3d\_Pre\_n.lp* for the pre processor
- *Opera3d\_Post\_n.lp* for the post processor.

User input is written to files called

- *Opera3d\_Modeller\_n.log* for the modeller
- *Opera3d\_Pre\_n.log* for the pre processor
- *Opera3d\_Post\_n.log* for the post processor

*log* files are in a format which can be used as input to the program with the **\$ COMINPUT** command. Graphical input or cursor commands are included in the log files; they can be read back into the program when the **\$ COMINPUT** command is accessed from the GUI. (**\$ COMINPUT** is described on [page 2-22](#).) The pre processor also written to a file called *Opera3d\_Pre\_n.backup*. This file can be used in place of a pre processor data file (*.oppre*). A unique set of files is created for each run of the programs. The lowest available value of **n** is used for all files.

*lp*, *log* and *backup* files are stored in a sub-folder (sub-directory) of the project folder (current directory) named *opera\_logs*.

Additional output files can be created by the user to contain the program's usual output or user-defined output or both. The commands to do this are described on [page 2-24](#).

## Commands and Parameters

---

Commands and parameters control the programs. All commands and their parameters may be shortened to their minimum unambiguous form. In any case, a maximum of 16 characters is used, except for file name parameter values. For example, in order to input the **DISPLAY** command, any of the following character strings could be typed: **DISPLAY** or **DISPLA** or **DISPL** or **DISP** or **DIS** or **DI**. The single character **D** will not be sufficient, because other commands also begin with this character. The programs' command interpreter will return an intelligible message when an error is detected in the input. If **D** had been input as a command the interpreter would reply:

```
DCOD Message 2: Command 'D' ambiguous (CMND)
```

## The Help Character (!)

---

Short help messages on the commands and their parameters can be obtained at any time by entering the help escape character; this is the exclamation mark (!). Entering a single exclamation mark on a new line will cause a list of all the commands to be displayed, together with a one line description of each command's function. Entering a command name followed by a single exclamation mark will produce a one line description of the command, followed by a list of all the command's parameters with their current value and a description of their function.

- Examples: (Not all the commands are shown here.)

```
OPERA > !
Valid commands are:
BHDATA  Define and modify BH data
SOLVE   Create or update an analysis database
READ    Read a file of OPERA-3d pre processor data
END     End OPERA-3d pre processor
$...    Built-in commands. Type '$ !' for a list.
OPERA > read !
Read a file of OPERA-3d pre processor data
Parameter Value Meaning
FILE          File name
```

## Parameter Assignment

---

Parameter values are specified either by entering an assignment instruction

**parameter=value**

or positionally by entering the values for the parameters in sequence. Both forms of specification may be mixed, in which case specifying

**key=value1 value2 value3**

implies that **value2** is assigned to the next parameter after **key** and **value3** to the one after that. The parameter sequence for a command is fixed in the order listed by the help escape character **!**. When assignment instructions are used to specify the value of parameters the order is not important, except when expressions which reference other parameters are used (see examples on [page 2-10](#)).

Parameter assignments may be separated either by a comma or spaces; any number of spaces may be used, but if two commas are used in positional input mode this implies that the parameter value is not supplied. Whichever input mode is being used, a comma at the end of an input line implies that the command will be continued on a subsequent line. In this case the parameters entered on the first line are assigned in the program, but the action is not initiated. The first parameter on a continuation line must be assigned explicitly, i.e. using **parameter=value** syntax.

Parameters are unique to the command with which they are associated. The only exceptions to this are the parameters **COMPONENT**, **VX**, **VY** and **VZ**. These take expressions to define the output field quantities. In this case the expression(s) given in one command become the default values for other commands which use the parameters.

The value of the parameters associated with a command are in general initialized to sensible defaults when the programs start, although there are cases where it is not sensible to provide a default. For example, there is no default for the file name with the **READ** command. The last value used for a parameter (in a command) becomes the default value for that parameter the next time the command is used, except in cases where this could be disastrous. The exceptions are obvious, for example, with commands that delete objects the object names or numbers will not be defaulted to the last value.

- Example: Using the following **DISPLAY** command as an illustration:

```
OPERA > display !
Display mesh and conductors
Parameter Value Meaning
SIZE      100.0 The size of the viewing window
XEYE      0.0   X coordinate of eye position
YEYE      0.0   Y coordinate of eye position
ZEYE      0.0   Z coordinate of eye position
```

- Example: Assignment instruction mode:

```
OPERA > disp size=50 xeye=10 yeye=5 zeye=2
```

- Example: Positional input mode

```
OPERA > disp 50 10 5 2
```

- Example: Mixed positional and assignment

```
OPERA > disp xeye=10 5 2 size=50
```

- Example: Missing positional input, **YEYE** takes last value by default

```
OPERA > disp size=50 xeye=10,,2
```

## Parameter Values

---

There are 4 types of value which may be assigned to a parameter: **Numeric**, **Expression**, **Character** and **Boolean**. Some parameters can take several value types but some combinations, such as character and expression are not allowed. Error messages indicate if an inappropriate value type has been used, e.g.

DCOD Message 19: Parameter 'TYPE' cannot take numeric values (DECODE)

### Numeric Parameter Values

Numeric values are used in many commands for specifying position, size, number of objects etc. Numeric values can be integer, fixed or floating point REAL or DOUBLE PRECISION numbers.

- Examples:

23  
1.2  
3E5  
-5.789E+04  
2.305983743795d5  
-0.04

### Expressions in Parameter Values

Most parameters which can take numeric values can also take algebraic expressions to specify the values. Expressions used in this way for data input are a replacement for a calculator. Variables within such expressions can be other parameters, system variables or user variables (see the **\$ PARAMETER** and **\$ CONSTANT** commands on [page 2-19](#)). These input expressions are not remembered; they are evaluated and the result is stored.

Parameters which cannot take expressions as values are those which also take character values. The text functions, **%INT()** and **%REAL()** ([page 2-11](#)) provide a way of getting around this restriction.

Expressions are also used to specify user defined parameters for output field quantities in post processing. These expressions are remembered and used for evaluation when referenced. Variables in output expressions can also include the position and the field components. Full details are given in Chapter 5.

Within expressions, variable (parameter) names cannot be abbreviated. If a command parameter is used in an expression, its name must be typed in full.

The following characters can be used in expressions, with their usual FORTRAN meanings: + - / \* ( ). No spaces can be included within expressions.

The following functions are supported, again using their usual FORTRAN definitions:

<b>FUNCTIONS</b>	
<b>Arithmetic</b>	
<b>ABS ( a )</b>	the modulus of <b>a</b>
<b>INT ( a )</b>	the largest integer whose magnitude does not exceed the magnitude of <b>a</b> times the sign of <b>a</b>
<b>NINT ( a )</b>	the integer closest to <b>a</b>
<b>MAX ( a ; b )</b>	the maximum of <b>a</b> and <b>b</b>
<b>MIN ( a ; b )</b>	the minimum of <b>a</b> and <b>b</b>
<b>MOD ( a ; b )</b>	the remainder when <b>a</b> is divided by <b>b</b>
<b>SIGN ( a ; b )</b>	the modulus of <b>a</b> times the sign of <b>b</b>
<b>Trigonometry (angles in radians)</b>	
<b>ACOS ( a )</b>	the angle whose cosine is <b>a</b>
<b>ASIN ( a )</b>	the angle whose sine is <b>a</b>
<b>ATAN ( a )</b>	the angle whose tangent is <b>a</b> , angle in the range $\left(-\frac{\pi}{2}, \frac{\pi}{2}\right)$
<b>ATAN2 ( a ; b )</b>	the angle whose tangent is <b>b/a</b> taking into account the signs of <b>a</b> and <b>b</b> and allowing <b>a</b> to be zero, angle in the range $(-\pi, \pi)$
<b>COS ( a )</b>	the cosine of <b>a</b>
<b>COSH ( a )</b>	the hyperbolic cosine of <b>a</b>
<b>COTAN ( a )</b>	the cotangent of <b>a</b>
<b>SIN ( a )</b>	the sine of <b>a</b>
<b>SINH ( a )</b>	the hyperbolic sine of <b>a</b>
<b>TAN ( a )</b>	the tangent of <b>a</b>
<b>Trigonometry (angles in degrees)</b>	
<b>ACOSD ( a )</b>	the angle whose cosine is <b>a</b>
<b>ASIND ( a )</b>	the angle whose sine is <b>a</b>
<b>ATAND ( a )</b>	the angle whose tangent is <b>a</b> , angle in the range $(-90, 90)$

<b>FUNCTIONS (continued)</b>	
<b>ATAN2D (a;b)</b>	the angle whose tangent is $b/a$ taking into account the signs of $a$ and $b$ and allowing $a$ to be zero, angle in the range $(-180, 180)$
<b>COSD (a)</b>	the cosine of $a$
<b>SIND (a)</b>	the sine of $a$
<b>TAND (a)</b>	the tangent of $a$
<b>Exponentials and logarithms</b>	
<b>EXP (a)</b>	the value of $e^a$
<b>LOG (a)</b>	the natural logarithm of $a$
<b>LOG10 (a)</b>	the common logarithm of $a$
<b>SQRT (a)</b>	the square root of $a$

**N.B.** Functions with 2 arguments, `ATAN2` and `MOD`, use ‘;’ to separate the arguments, since ‘,’ is the separator between parameter assignments.

- Example: Parameter assignments:

```
OPERA > line x1=x1+10 x2=x2+10
```

- Example: order is important since the expressions are decoded in the order given; the following commands are not equivalent:

```
OPERA > line x1=y1+4 y1=y1+3
```

```
OPERA > line y1=y1+3 x1=y1+4
```

- Example: Output components in post processing:

```
OPERA > map comp=sqrt(x**2+y**2)*bx
```

## Character Values for Parameters

Character values are character strings, starting with an alphabetic character. In most cases the value is compared against a list of valid options. In such cases the value can be abbreviated to its minimum non-ambiguous length. Specifying the help character, ‘!’, will cause the program to give a list of the valid options.

In other cases character values are used to give file names. In such cases longer strings are permitted. For operating systems which allow file name extensions or file types, the types are added automatically, the precise type being determined from the context. On systems where file names are case sensitive, file names which are entirely upper-case are given upper-case extensions; other file names are given lower-case extensions.

File names given as tree-names can include environment variables within the directory part of the name. Variables `$VFDIR` (UNIX) and `%VFDIR%` (Windows) are defined by the software as the parent directory or folder holding the software.

Some character strings are used for titles or text messages. For these strings, the rule about the first character being alphabetic can be relaxed. However any string which contains spaces or commas must be enclosed in quotation marks (`'`). Quotation marks embedded within character strings must be paired. The GUI automatically supplies quotation marks when necessary.

In the pre processor, superscripts can be used in title strings. This is done by including the characters `~E` before and `~A` after the superscripts. It is not necessary to use `~A` if the string ends with superscripts.

- Examples:

```
OPERA > disp label=pote
OPERA > acti file=quadrupole
OPERA > title string='Field after 10~E-3~As'
```

The last example would give the title:

```
Field after 10-3s
```

## Boolean Parameter Values

Boolean parameters take the values YES or NO and are in general used for switching features on or off. Boolean values can also be specified by `+PARAMETER` or `-PARAMETER`, being equivalent to `PARAMETER=YES` and `PARAMETER=NO`.

Some parameters can take boolean or character values

- Examples:

```
OPERA > disp +perspective
OPERA > disp pers=yes
OPERA > disp elem=no
```

## Text functions

It is sometimes necessary to insert the value of an expression into a parameter value as a character string, for example, to include an index number in a file name or to supply a value by expression to a parameter which cannot take expressions. This can be done using two text functions:

**%INT**(*expression*)

**%REAL**(*expression*)

These functions evaluate the expressions given and replace *%function(expression)* on the command line with characters representing the value (**%REAL**) or the nearest integer to the value (**%INT**).

For example, to specify the maximum and minimum contour values using user variables:

```
disp min=%real(#min) max=%real(#max)
```

Similarly, the text function

**%EXPR**(*variable*)

replaces itself with the expression which *variable* represents. The *variable* can be the parameter of a command, e.g. **COMP** or a user variable. It can be used to modify the expression. In the following example, the first component expression is modified by dividing by a constant:

```
disp comp=bmod/hmod  
disp comp=%expr (comp) /mu0
```

This second command is equivalent to

```
disp comp=bmod/hmod/mu0
```

## Command Interpreter Errors

---

The command interpreter provides input error recovery facilities. If a parameter name is mistyped, the other assignments on the input line will be performed, unless they are positional assignments whose position can not be determined, but command action will not continue. The incorrect parameter(s) can then be re-specified without having to retype the whole input line. The same applies to errors detected in the value of a parameter. The command interpreter will display any portion of the input string which it cannot recognize or which it thinks is in error so that the user can see which parameters need to be re-specified.

## Confidence Level

---

Experienced users rely on the last used defaults and the mixed assignment and positional input modes to make efficient use of the programs. There are other useful features in the interpreter which can be used to reduce the amount which has to be typed.

### Repeated commands

If the same command is being repeated many times the command name need not be supplied, providing that an assignment instruction starts the input line.

- Examples:

```
OPERA > disp size=100
OPERA > size=10
```

### Prompted input of parameter values

The final feature of the command processor is its prompted input mode. Issuing a command followed by two help escape characters (!!) puts the command interpreter into prompt mode. Each parameter is displayed together with its default value and description. The default is accepted by pressing the **<Enter>**, **<Return>** or ↵ key, or a new value may be entered. When all the parameters have been offered the program waits for either **<Enter>**, **<Return>** or ↵ to be pressed, which then executes the command, or if '\$ABORT' is entered the command is aborted. '\$ABORT' can be used instead of any parameter value to abort the prompting at that point and not execute the command. '\$SKIP' can be used to skip over the remaining parameters and execute the command.

Note that Boolean parameters cannot be specified using **+PARAMETER** or **-PARAMETER** when in prompt mode. The character values **YES** and **NO** should be used instead.

- Example:

```
OPERA > disp !!
There are 24 parameters
For each parameter:
hit return to accept default
OR enter new value
OR type $HELP for help
```

OR type \$SKIP to skip remaining parameters and execute command

OR type \$ABORT to skip remaining parameters and abort command

NO. Name Value Meaning

1 SIZE 100 Size of half-axis in each direction

!! > 10

2 XEYE 0 X coordinate of eye position

!! > 3

3 YEYE 0 Y coordinate of eye position

!! > 4

4 ZEYE 1000 Z coordinate of eye position

!! > 5

Type return to obey command, or \$ABORT to abort

!! >

## Prompted free format Input

---

Once a specific option has been selected by command or graphical input the programs may prompt for extra input to define further parameters. In such cases the user is shown the parameters required and asked to provide values. The parameters are input in free format using **<space>** or comma as the parameter separator. The order of the parameters in this type of input is shown by the prompt, however parameters defined in the manuals as optional keywords may be specified in any order. Free format input lines cannot be continued on subsequent lines by means of a comma.

In some contexts, for example coordinate input in the Point Definition Mode of the pre processor, values can be omitted in free-format input so that default, or previously specified values apply. The first value is omitted by using a comma at the start of the input line; subsequent values are omitted by using repeated commas within the line or by truncating the line after one or more values.

- Example: Boundary condition input in the pre processor **DEFINE** command.

```
Define the boundary condition for this face e.g. POT 0.0
OP-B/C > pot 2 all
```

The following line would be rejected since the keyword potential is not followed by a value.

```
OP-B/C > potential all 2
```

So that expressions can be used in free format input, each item is given a name according to its position on the line. #1 is the first; #2 the second etc. These names can be used to access default values in the input of coordinate positions or to use the values of earlier items on the same line.

- Example: Construction line input, the value of  $u_2$  is set in terms of  $u_1$ :

```
OP-C/LINES > line 1.34*sin(pi/6) 10 #2+20 10 0
```

## Built-in Commands

---

‘Built-in’ commands provide control-structures (loops and conditions), user variables, command input from files and access to the operating system. ‘Built-in’ commands can be used at (almost) any prompt.

**\$** at the start of an input line introduces a ‘built-in’ command or **\$**-command. There is a built-in dictionary of commands and parameters and the normal

```
$ command parameter=value ...
```

syntax can be used. The parameters have been ordered so that it is natural to use positional assignments. The parameter names are useful to provide on-line documentation using the **!** character. Except where noted, there are no default values. **\$**-commands must be specified in full and cannot be continued on subsequent lines.

### Limitations

The code which implements these loops and control-structures has the following limitations. Control structures can be nested to a depth of 20 levels. The maximum number of commands from the first control command to the last **\$ END** (inclusive) is 10000.

### Loops

Three types of loop are available: **\$ DO**, **\$ FOR** and **\$ WHILE**. In each case the commands between the loop command and the corresponding **\$ END** command are executed a number of times.

#### **\$ DO-loops**

The **\$ DO**-loop is similar to the FORTRAN do-loop. At the start of each execution of the loop, an index-variable is set to a value specified by a starting value, a final value and an increment. The syntax of the command is

```
prompt > $ DO index start final increment
... commands to be executed ...
prompt > $ END DO
```

The *index* should be the name of a user-variable, (up to 5 characters, beginning with #). Its value can be changed within the loop, but is always set to the correct value (c.f. **\$ CONSTANT**, see [page 2-19](#)) at the start of the loop.

*start*, *final* and *increment* can be specified as numerical values or expressions. Expressions are evaluated before the first pass through the loop. If *increment* is omitted it has a default value of 1.

### **\$ FOR**-each loops

The **\$ FOR**-each loop executes a set of commands with a user-variable set in turn to each of the expressions given on the **\$ FOR** command. The syntax is

```
prompt > $ FOR index ex1 ex2 ex3 ... ex9
... commands to be executed ...
prompt > $ END FOR
```

At least one, and at most 9, expressions ( $ex_n$ ) can be given. *index* is assigned in turn to each expression (c.f. **\$ PARAMETER**, see [page 2-19](#)) at the start of the loop.

### **\$ WHILE**-loops

The **\$ WHILE**-loop executes a set of commands while a logical expression remains true. The syntax is

```
prompt > $ WHILE ex1 logical_operator ex2
... commands to be executed while logical expression is true ...
prompt > $ END WHILE
```

The logical operators are **EQ**, **NE**, **LE**, **LT**, **GE** and **GT**.

## Conditional commands

Three conditional commands are available: **\$ IF**, **\$ ELIF** and **\$ ELSE**. The commands **\$ IF** and **\$ ELIF** should be followed by a logical expression. The **\$ ELSE** command has no parameters.

A **\$ IF** block (the commands executed if the logical expression is true) is terminated by a **\$ ELIF**, **\$ ELSE** or **\$ END IF** command.

A **\$ ELIF** (else-if) block is terminated by a **\$ ELIF**, **\$ ELSE** or **\$ END IF** command; it is only executed if the logical expression is true and none of the preceding **\$ IF** or **\$ ELIF** blocks at the same level have executed.

A **\$ ELSE** block is terminated by a **\$ END IF** command; it is only executed if none of the preceding **\$ IF** or **\$ ELIF** blocks at the same level have executed.

The syntax is

```
prompt > $ IF value1 logical_operator value2
... commands to be executed if logical expression is true ...
prompt > $ ELIF value1 logical_operator value2
... commands to be executed if previous blocks have not been executed and logical
expression is true ...
prompt > $ ELSE
... commands to be executed if previous blocks have not been executed ...
prompt > $ END IF
```

The logical operators are **EQ**, **NE**, **LE**, **LT**, **GE** and **GT**.

## The \$ END command

The **\$ END** command ends the current block (**DO**, **FOR**, **IF** or **WHILE**). Although the block type is not logically necessary, it must be specified to ensure that the user knows which block is being **ENDED** and to help in supplying the correct number of **\$ END** commands. The syntax is

```
prompt > $ END block_type
```

## User Variable Commands

The **\$ CONSTANT**, **\$ PARAMETER** and **\$ STRING** commands define user variables. Two further commands, **\$ ASK** and **\$ PROMPT**, request the user to supply values for user variables and are described on [page 2-23](#).

Each of the three commands has the same two parameters. The first defines the **NAME** of the user variable and the second the **VALUE**. If the name is used again then the value for that variable is overwritten. If no **VALUE** is given, the current value for the **NAME**d user variable is displayed. If **NAME=!** is used then all the user variables currently defined are listed.

The second parameter gives the **VALUE** for the user variable.

### Numerical Variables

Numerical user variable names, defined with **\$ CONSTANT** or **\$ PARAMETER**, start with **#** and have up to 16 characters.

The **VALUE** can be a simple numeric value or can be an expression referencing other user variables or system variables. The **\$ CONSTANT** command evaluates the **VALUE** at the time the command is used and any expression is lost. The

**\$ PARAMETER** command stores the expression given by the **VALUE** parameter so that it can be re-evaluated each time the variable is referenced.

- Example - to define degrees to radians conversion factor:

```
OPERA > $ cons #fac pi/180
Assign a value to a user variable
Name Value      Expression or Value
#FAC  0.0174533  0.0174533
OK
```

It is possible to write simple programs using the **\$ CONSTANT** and **\$ PARAMETER** commands. The user parameters are evaluated at the time they are defined and again whenever they are referenced. Thus changing a user variable definition implies a change in all user parameters which reference that variable. This is shown by the following example. Note how changing the value for #A implies a change in value for #B and #C.

- Example

```
OPERA > $ cons #a 3
Assign a value to a user variable
Name Value  Expression or Value
#A   3.0    3.0
OK
OPERA > $ para #b #a**2
Assign an expression to a user variable
Name Value  Expression or Value
#B   9.0    #A**2
OK
OPERA > $ para #c #b-4
Assign an expression to a user variable
Name Value  Expression or Value
#C   5.0    #B-4
OK
OPERA > $ cons name=! value=0
Assign an expression to a user variable
Name Value  Expression or Value
#A   3.0    3.0
#B   9.0    #A**2
#C   5.0    #B-4
OK
```

```

OPERA > $ cons #a 2
  Assign a value to a user variable
Name Value Expression or Value
#A    2.0    2.0
OK
OPERA > $ cons n=!
Assign a value to a user variable
Name Value Expression or Value
#A    2.0    2.0
#B    4.0    #A**2
#C    0.0    #B-4
OK

```

*Menu Routes to*  
**\$PARAMETER**  
*and*  
**\$CONSTANT**

```

OPTIONS↓
  Parameters

OPTIONS↓
  Constants

Options
  User variables

```

The **\$ PARAMETER** and **\$ CONSTANT** commands are also available at many places in the pre and post processor GUI using the menu option Calculator.

*Character*  
*Variables*

Character variable names, defined with the **\$ STRING** command, have up to 16 characters, starting with a letter.

The **VALUE** can be any character string. The character string can be recovered on (almost) any input line by use of the **NAME** surrounded by ampersands (&). Any quotation marks used to define the string are lost. This allows several strings to be concatenated.

There are two predefined character variables, **NOW** and **TODAY**, which can be used to obtain the current time and date.

- Example - storing a title for later use (note the use of `%real` to obtain a character representation of the value of a system variable (see [page 2-11](#)):

```

OPERA > $ string t1 'Septum Magnet'
Assign a string to a user variable
Name String
T1      Septum Magnet
OK
OPERA > $ string t2 '(RMS error %real(#err)%)'
Assign a string to a user variable
Name String
T2      (RMS error 5.23146%)
OK
OPERA > title '&t1& &t2&' tr
Set a title for the graphics window

```

The title displayed is 'Septum Magnet (RMS error 5.23146%)'.

## Command Input Files

The `$ COMINPUT` command allows commands to be read from a file and additionally sets the message output mode. If a file with no file name extension is given, the extension `comi` is assumed. The syntax is:

```
prompt > $ COMI filename mode
```

If the `$ COMINPUT` command appears in a loop, the file of commands is read each time the loop is executed. Almost any commands can be included in command input files.

### Menu Route to `$ COMINPUT`:

```

FILE↓
  Commands in

```

### Text Output Modes

The parameter `MODE` applies whether or not a command file is requested. In `PAGED` and `CONTINUOUS` modes, the text output continues until the next input is requested, and with `MODE=OFF` most of the normal text output does not appear at all. `MODE=PICTURES` is useful for running 'demonstration' command files, since the program pauses for an `<Enter>`, `<Return>` or `↵` before each time the graphics window is cleared, but does not stop when the text window is full. In each `MODE`, text output is written to the dialogue file.

When menus are being used the text output modes are slightly different. While a command file is being read with **MODE=CONTINUOUS**, text output appears on the text window and does not appear in GUI MessageBoxes. An additional option, **MODE=MESSAGE** causes the GUI MessageBoxes to be used.

### *Menu Route for Text Output Mode:*

**FILE**↓  
Commands in

Execution of command files can be interrupted using the settings of the **MODE** parameter. It can also be interrupted by inclusion of **\$ PAUSE** commands. The syntax is:

```
prompt > $ PAUSE seconds
```

**\$ PAUSE** waits for a number of seconds before continuing. If *seconds* is omitted or is  $\leq 0$ , the program waits for the user to type **<Enter>**, **<Return>** or **↵** or dismiss a MessageBox before continuing.

### *Prompting Commands*

Command input files can contain user variables, which must be assigned values before the commands are executed. The **\$ ASK** command can be used to request the user to supply a value for a numerical variable (c.f. the **\$ CONSTANT** command). The **\$ PROMPT** command can be used to request the user to supply a value for a character variable (c.f. the **\$ STRING** command). The syntax is:

```
prompt > $ ASK #name prompt_string
prompt > $ PROMPT name prompt_string
```

The optional *prompt\_string* is displayed to show what input is required. The value must be supplied at the keyboard before the program will continue.

N.B. **\$ ASK** and **\$ PROMPT** cannot be used in OPERA-3d pre processor data files; they can only be used in command input files.

### *File Existence Command*

The **\$ EXIST** command tests the existence of a file. The syntax is

```
$ exist filename
```

If the file exists, the system variable **FILEEXISTS** is given the value 1; if the file does not exist, **FILEEXISTS** is set to 0.

### *Error Handling Commands*

The **\$ ERRORHANDLER** command selects the behaviour of the command processor after an error in a **\$**-command has been detected. The default behaviour, which can be selected using **\$ errorhandler yes**, is that all commands already stored for execution are ignored.

If **\$ errorhandler no** has been specified, the programs continue to execute stored commands. (Commands are stored during execution of **\$ COMINPUT**, **\$ DO**, **\$ FOR** and **\$ WHILE** commands.)

The **\$ BREAKERROR** command causes the command processor to exit the current loop if an error has been detected. This enables command loops which read (see [page 2-24](#)) to the end of a file without knowing in advance the number of lines in the file. A typical sequence of commands could be:

```
$ constant #i 1
$ while #i eq 1
$ read ...
$ breakerror
other commands
$ end while
```

## User Input/Output Commands

There are seven commands for user input and output of files. Before a file can be read or written it must be opened.

```
prompt > $ OPEN stream filename authority ± REDIRECT
```

opens a file on a logical stream number which can be in the range 1 to 10. The file can be used in 4 ways, depending on the **authority**. These are **READ** an old file, **WRITE** a new file, **OVERWRITE** an old file and **APPEND** to an old file.

If **+REDIRECT** is selected for an output file, the output which is written to the dialogue file will be written to the output file as well.

When all input or output has been completed a file can be closed to release its logical stream number or to make it available for opening with different authority.

```
prompt > $ CLOSE stream
```

closes a logical stream number in the range 1 to 10.

The **\$ READ** command takes one line from the file opened on the given logical stream number and assigns any numerical data on the line to user variables. Up to 20 variable names can be given. Any character strings on the line are ignored. The syntax is

```
prompt > $ READ stream #var1 #var2 #var3 ...
```

The **\$ WRITE** command is similar to the **\$ READ** command. The syntax is

```
prompt > $ WRITE stream data1 data2 ... ±OUTPUT
```

Up to 20 data items can be supplied.

A line of output can be built up using several **\$ WRITE** commands. If this is necessary the first **\$ WRITE** commands should have **-OUTPUT**. The last **\$ WRITE** should have **+OUTPUT**. The data from the second and subsequent **\$ WRITE** commands will be positioned after the data of the previous write commands in an internal buffer which is written and re-initialized when **+OUTPUT** is used.

Data items on a **\$ WRITE** command can be numerical, characters or user variables. Before the **\$ WRITE** command is used, the **\$ FORMAT** and **\$ ASSIGN** commands should be used to define the type of data and the style of output to be used for each item on the **\$ WRITE** command.

The **\$ FORMAT** command can be used to define up to 20 different formats for output items. In each form the *width* can be specified as zero which implies that the program should calculate a width to fit the data being written. The syntax has one of the following forms.

- To define a format for character data, truncated or padded with spaces to a particular **width**:

```
prompt > $ FORMAT number CHARACTER width
```

- To define a fixed point format for numerical data, with **decs** as the number of decimal places.

```
prompt > $ FORMAT number FIXED width decs
```

- To define a floating point format for numerical data:

```
prompt > $ FORMAT number EXPONENTIAL width
```

- To define a format for integer data:

```
prompt > $ FORMAT number INTEGER width
```

- To define a format for a user variable to display the expression defining the variable truncated or padded to a particular **width**:

```
prompt > $ FORMAT number USER width
```

- To define a character string to be output irrespective of the data on the **\$ WRITE** command (N.B. It is necessary to define **STRING** formats containing spaces to appear between other data items if those other data items are written with a width of zero.):

```
prompt > $ FORMAT number STRING width STRING=chars
```

- To list the defined formats:

```
prompt > $ FORMAT +LIST
```

The **\$ ASSIGN** command assigns format numbers to the data items of subsequent **\$ WRITE** commands. The first data item will be written with the first non-**STRING** format, the second item with the next non-**STRING** format, and so on. The syntax of the **\$ ASSIGN** command is:

```
prompt > $ ASSIGN form1 form2 form3 ...
```

Up to 20 formats can be assigned.

The final input/output command is **\$ BACKSPACE** which allows a file opened with **\$ OPEN** to be backspaced or positioned at its start. The syntax is:

```
prompt > $ BACKSPACE stream records
```

where *stream* is the logical stream number and *records* is the number of records the file is to be backspaced. If *records* is given as -1, the file is positioned at its start.

- Example: reading and writing files.

A file *points.dat* contains:

```
10
X 0 Y 0
X 0 Y 1
X 0 Y 3
X 0 Y 5
X 0 Y 7
X 0 Y 9
X 0 Y 10
X 1 Y 1
X 1 Y 5
X 1 Y 9
```

The following commands will read the data and use the coordinates in **POINT** commands to evaluate the field and to write a corresponding output file. Note the use of comments.

```
/ Open input file for reading
$ open 1 points.dat read
/ Read the first line into user variable #np
$ read 1 #np
/ Open output file for writing
$ open 2 fields.dat write
/ Define a string format to space the output
$ form 1 string string=' '
/ Define a floating point format
$ form 2 expo 0
/ Assign format numbers for the output
$ assi 2 1 2 1 2 1 2
```

```

/          Start a loop from 1 to #np
$ do #i 1 #np
/          Read #x and #y from input file
$ read 1 #x #y
/          Evaluate fields at #x #y
poin x1=#x y1=#y
/          Write coordinates and flux density to output file
$ write 2 x y bx by
/          End of loop
$ end do
/          Close files
$ close 1
$ close 2

```

The output file, *fields.dat* contains:

```

0.0 0.0 1.179564E-07 -0.000120992
0.0 1.0 6.322637E-08 -0.000121184
0.0 3.0 3.739035E-06 -0.000133139
0.0 5.0 -4.42901E-06 -0.00013248
0.0 7.0 -7.59639E-06 -8.48776E-05
0.0 9.0 -7.74227E-06 -2.32268E-05
0.0 10.0 0.0 0.0
1.0 1.0 4.347521E-07 -0.000121116
1.0 5.0 -1.34893E-05 -0.000141764
1.0 9.0 -3.10067E-05 -3.09315E-05

```

## Operating System Commands

There are two commands to execute operating system commands:

```

prompt > $ OS str1 str2 str3 str4 str5 str6
prompt > $ CD directory

```

**\$ OS** passes up to 6 strings ( $str_n$ ) which together form a command to the operating system to be executed. This can be used to issue single commands to list names of files in directories (folders), delete or rename files, etc., using the normal syntax of the operating system. The output from the commands is listed, with the usual page breaks.

### **The \$ OS** command on UNIX systems

On UNIX systems, in order to redirect the output from the command to a file, the program adds to the command the appropriate notation:

- in c-shell: **user\_command >& TeMpOsCmNdFiLe**
- in other shells: **user\_command > TeMpOsCmNdFiLe 2>&1**

The contents of the file, *TeMpOsCmNdFiLe*, are then listed. For this reason, shell meta-characters within the user command should be used with care and it might

be necessary to enclose the user command in parentheses. For example, to run a background command use the following syntax:

```
$OS (xterm &)
```

### **The \$ OS command on Windows systems**

On Windows systems a limited set of Command Prompt commands have been implemented, although without full functionality. The commands available are:

- **dir** or **ls** to list files in the current folder
- **del** or **rm** to delete a file
- **mkdir** or **md** to create a new folder in the current folder
- **ren** to rename a file
- **copy** to copy a file
- **pwd** or **cd** to report the current folder

**N.B.** File names including spaces should be enclosed in double-quotes (").

### **Additional \$ OS Commands to Launch OPERA**

On all operating systems additional commands are available for starting interactive and analysis programs from the interactive programs. The commands for the interactive programs are:

```
$ OS operapre
$ OS operapost
$ OS operamodeller
```

and for the analysis programs:

```
$ OS operaanl program datafile mode
```

where

- *program* is one of **CARMEN**, **ELEKTRASS**, **ELEKTRATR**, **ELEKTRAVL**, **SCALA**, **SOPRANOEV**, **SOPRANOSS** or **TOSCA**
- *datafile* is the name of the OPERA3d database to be analysed
- *mode* is **FORE** or **BACK**. The interactive program waits while **FORE**ground analysis jobs are run, but can be continued or ended while **BACK**ground jobs are run.

**The \$ CD  
command**

Because the \$ OS command spawns a new sub-process, a command such as  
prompt > \$ OS cd directory

has no lasting effect. The \$ CD *directory* command should be used instead to change the current directory or folder. Directory names can be given using environment variables. Environment variables \$VFDIR (on UNIX systems) and %VFDIR% (on Windows systems) are defined by the software as the parent directory of the software.

**Menu Routes:**

FILE↓  
OS command

FILE↓  
Change directory

## Command Separator and Comments

---

The command separator allows several keyboard commands to be given on one input line. The command separator is the vertical bar, |.

- Example using the **LINE** and **PLOT** commands:

```
OPERA > line 0 0 0 10 0 0 | plot comp=by
```

If the first non-space character on an input line is /, the line is treated as a comment. Comments are output to the dialogue file and comments in command files are displayed as the file is being read. In menu mode, comments which start /\* are displayed in GUI MessageBoxes.

In some contexts it is not possible to give comments, since for some operating systems, file names can begin with /.

## Euler Angles

---

The programs occasionally require that the orientation of a component or a physical property be defined. Euler angles are always used to define the orientations. The following table shows the Euler angle convention used:

ANGLE	DESCRIPTION
<b>PHI</b> or <b>P</b>	Rotation about the original (global) Z axis positive rotation by right-hand screw convention i.e. from X towards Y.
<b>THETA</b> or <b>T</b>	Rotation about the local Y axis created by the <b>PHI</b> rotation, positive right-hand screw convention i.e. from Z' to X'.
<b>PSI</b> or <b>S</b>	Rotation about the local Z axis created by the <b>PHI</b> and <b>THETA</b> rotations, positive right-hand screw convention i.e. from X' to Y'.

Unless there are parameter names, the convention is always to define the rotation by the ordered triple **THETA PHI PSI** or **T P S**.

**N.B.** This order is different from the order used in the definition above.

Examples:

Coordinate Transformation	$\vartheta$	$\Phi$	$\Psi$
$XYZ_{local} = XYZ_{global}$	0	0	0
$XYZ_{local} = YZX_{global}$	90	0	90
$XYZ_{local} = ZXY_{global}$	90	90	180

Whenever the ordered triple for Euler angles is to be entered into OPERA-3d Modeller or pre and post processors, the escape function **\$EULER** can be typed instead. The programs prompt the user to define the Euler angles by specifying **THREE** points or by **SWAPPING** the coordinate axes. The 3 points required are a local origin, and points on the local Z axis and either the local X or the local Y axis. The coordinate axes can be swapped by specifying the global axes which correspond to the local X and Y axes. The global axes, in either positive or negative directions, are referred to by keywords, **POSX**, **NEGX**, **POSY**, **NEGY**, **POSZ**, **NEGZ**. The Euler angles can be stored within OPERA-3d under user defined names and recalled later by name.

- Example: Setting material **VECTOR** property in the pre processor.

Set the material name, potential type and constants (or HELP or QUIT)

```
OP-MATERIALS > iron total vector $euler
```

Give name of Euler angle set. Type LIST for a list or NEW to define a new Euler angle set.

```
OP-EULER > new
```

Give name and definition method. Methods are:

THREe - coordinates of origin, and points on z and one other axis

SWAP - swap global axes

```
OP-EULER > iron swap
```

Give a combination of POSX, NEGX, ..., POSZ, NEGZ to define local X, Y and Z axes. (Only X and Y need be given.)

```
OP-EULER > posy posz
```

Input line edited to replace \$EULER:

```
iron total vector 90 0 90
```

# *Chapter 3*

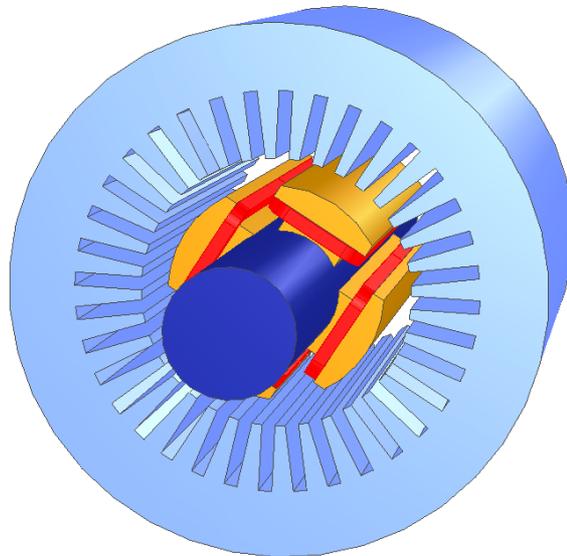
## **OPERA-3d Geometric Modeller**

### **Introduction**

---

The Geometric Modeller provides facilities for creating models for use with the OPERA-3d analysis modules and post processor.

The Modeller manipulates any defined objects through operations such as transformations and combinations. Basic objects (blocks, cylinders, spheres, cones, pyramids and toroids) can be created at any position in space. Once created they can be manipulated to reposition them. They can also be merged, intersected or subtracted from other objects in space to create more complex geometries.



**Figure 3.1 Complex geometry created with the Modeller**

This modelling technique allows many models to be created from the basic building blocks. Other more advanced techniques allow the geometry to be enhanced by e.g. sweeping an existing face.

## Reference Manual Overview

### *Geometric Modelling*

This section describes the terminology used throughout the program and guides, and introduces the basic techniques that are available when using the Modeller.

### *Reference*

This section introduces each of the commands available within the Modeller and gives details of the specific usage for each. Commands available through the menu system generate a command line equivalent. The details of how the commands operate is described in this section and gives more in depth detail of the functionality the command supplies.

# Geometric Modelling

---

## Topology Reference

This section describes the different terms used to describe the topology of a model.

### *Vertex*

This is a simple point in space, defined by a position within a cartesian coordinate system. Normally a vertex will mark a sharp change in geometry, e.g. the intersection of 2 or more edges.

### *Edge*

An edge is a line in space. Typically an edge will have 2 vertices defining its ends. The edge can also have an underlying geometry associated with it, e.g. it could be straight, an arc of a circle, or a more complex underlying spline curve. For certain edges, notably circles and ellipses, the edge may have a single vertex or no end vertices at all.

### *Face*

A set of edges, connected together at their ends, forms a loop. For most cases, such a loop forms the boundary of a face. A face also has an underlying geometry, e.g. the edges could bound part of a plane, part of a sphere or part of a more complex underlying surface definition.

There are also some cases where the face is bounded by more or less than a single loop. For some underlying surface geometries, e.g. a sphere or torus, there may be no bounding loop required. For structures formed from complex operations, there may be one or more internal loops that bound an interior section of the surface, excluding it from the face.

### *Cell*

A closed set of faces connected at the bounding edges forms a shell. A cell represents an enclosed volume of space bounded by one or more shells. Any point within the cell can be reached from another point within the cell by travelling a path that does not pass through a face. For some complex geometries there may be more than a single shell of faces bounding a cell, e.g. where a section has been removed from the interior of a volume, leaving an internal set of faces restricting the volume occupied by the cell.

### Body

A body is a collection of the above topological entities. All objects within the body are closely linked geometrically, so that parts of a body can not be moved without considering the impact on other parts of the body. For example a body may contain 2 cells. These cells each occupy their own space. Movement of one cell within the body may make the cells overlap which would cause the cells to become invalid.

However, if there are 2 bodies, each with one cell, it is permissible to have the bodies overlapping as there is no connection between the bodies. Before forming a final model, the bodies will be merged into a single body to ensure that the topology forms a single valid structure.

## Modeller Topology and Geometry Functionality

### Creating New Objects

Basic primitive commands exist for creating objects. The type of objects that can be created are:

Type	Command	Description
Block	<b>BLOCK</b>	Creates a cuboid block. Specify one corner and its opposite to define a cuboid.
Cylinder, cone, ellipsoidal cylinder	<b>CYLINDER</b>	Specify the positions at the centre of the base and centre of the top as well as the radius. Ellipsoidal cylinders and cones can be created by giving major / minor axes or a top radius of zero.
Sphere	<b>SPHERE</b>	Give the centre of the sphere and the radius
Torus	<b>TORUS</b>	Give the centre, the major and minor radii
Regular n-sided prisms and pyramids	<b>PRISM</b>	Creates a regular $n$ -sided polygon with corner points on an ellipse (of given major and minor radius), and extends this in the third dimension a given height with a taper given by a major radius at the top

These objects are created in the current Working Coordinate System. Local Coordinate Systems can be defined, and one of them selected to be the Working Coordinate System. All primitives are created within the Working Coordinate System, as are transformations.

A standard set of conductors can also be created. This is done through separate commands, e.g. the **RACETRACK**, **BEDSTEAD** commands. These conductors do not form a physical part of the model and are not included within the finite element mesh. They can be selected for transformation, repositioning and copying.

For more details see the **CONDUCTOR** section and the individual **CONDUCTOR** commands.

### *Modifying Objects*

Once created, objects within the model can be picked for modification. There are 2 ways to pick objects:

- by graphical interaction, or
- by picking objects that have certain attributes associated with them.

Once an object is picked, operations can be performed upon that object. In many cases it is possible to pick more than one object and to operate upon all of the picked objects.

The operation may perform differently depending upon the type of object picked, or may have no effect if the picked objects are not of suitable type.

For example: the copy operation (see the **TRANSFORM** command) can be applied to all picked bodies to create transformed copies of these bodies. If a cell has been picked, this can only be copied individually, and the process creates a new body containing the single cell that has been copied. A new body is created to ensure that the body's topology is not corrupted which may happen if the cell were copied into a space already occupied by another part of the body. A face will behave in the same way as the cell, but creates a 2d body that has no volume. Edges and vertices are not operated upon.

The topological entities that are affected by each command are listed in detail in the reference manual.

### *Graphical Interaction*

Graphical interaction is controlled by the **FILTER** command. This determines what type of topological object can be picked by positioning the cursor over it and double-clicking with the left mouse button. The options for the filter command include: vertex, edge, face, cell, body.

As well as the topological entities described, conductors and Local Coordinate Systems can be picked and modified.

When double clicking, the system generates a command to pick a numbered entity. If using the log file to generate command input files for automated modelling, commands generated through the graphical interaction should be avoided. Although these commands are valid, there is no guarantee that variations between platforms or versions of the software will produce the same numbering sequence. This may lead to incorrect models being produced. Where possible, picking by

attribute should be the preferred method when automating the Modeller using scripts.

### *Picking By Attribute*

Each entity can have attribute data associated with it. These data will vary according to the entity type. For example, bodies can have a name associated with them, and a set of labels that can be applied. Cells have a material label, a volume label, a potential type, an element type, a mesh control size and a set of labels. Faces have a boundary condition label, a mesh control size and a set of labels. Edges and vertices have only a mesh control size and a set of labels.

The **PICK** command can use these attributes and labels to identify parts of a model for modification, viewing, etc.

### *Modifying Objects Geometry*

The main command for modifying the geometry is the **TRANSFORM** command. This command can be applied to bodies. A limited form can also be applied to conductors and Local Coordinate Systems. All transformations occur within the Working Coordinate System. Transformations available include:

<b>Transformation</b>	<b>Description</b>
<b>TRANSLATION</b>	Re-position by translation
<b>ROTATION</b>	Rotation about an axis vector (through the origin) by an angle, or rotation through a set of Euler angles
<b>REFLECTION</b> (body only)	Reflect in a plane whose normal is specified (through the origin)
<b>SCALE</b> (body only)	Scale the body by anisotropic scale factors

### *Creating Complex Bodies*

More complex structures can be formed by performing Boolean operations upon 2 or more bodies using the **COMBINE** command. The Boolean operations are:

<b>Operation</b>	<b>Description</b>
<b>UNION</b>	Merge the 2 bodies so that the resulting body has the components that existed in both original bodies
<b>INTERSECTION</b>	Intersection will leave a body filling the volume of the model that was originally common to both bodies
<b>SUBTRACTION</b>	Removes volume from the first body that also lies within the second body

Operation	Description ( <i>continued</i> )
TRIM	As subtraction, but removes a copy of the second and subsequent bodies from the first picked body
CUTAWAY	Leaves the first body unchanged and cuts it from all other picked bodies

For **SUBTRACTION**, **TRIM**, and **CUTAWAY**, the order of picking bodies is important. In this case the second body picked is subtracted from the first body picked. If more than two bodies have been picked, the third body picked is subtracted from the result of the first subtraction, and so on. For other operations, the order is unimportant.

For all operations, there is an option to regularise the result. Regularising removes internal faces and edges that are formed from the operation. This is most easily seen during union, but can also be useful for intersection or subtraction.

### *Extending Geometry*

An existing planar face within the model may be picked and swept through space to extend the volume occupied by a body. When sweeping there are 3 options available:

Sweep operation	Description
DISTANCE	Sweep the face by a fixed distance normal to the surface
VECTOR	Sweep the face through a specified vector
ROTATION	Sweep the face around an axis, passing through a point, by a given angle

An option exists to keep the original face, in which case the sweep operation will form a new cell within the body.

The operation can also apply a taper so that the face expands or contracts at a constant angular rate.

When sweeping, care should be taken to avoid self intersections and formation of incorrect topologies. Where possible the sweep operation will try to fix these inconsistencies, but it cannot be guaranteed to work in all cases.

### *Blending And Chamfering*

The edges between adjoining faces can be blended to form a smooth interface between the faces, or chamfered to trim the edge. Both of these are achieved using the **BLEND** command.

Such blending and chamfering may only be applied for non-manifold geometry.

### *Removing Geometry Features*

Sections of the model can be deleted. Any picked body, conductor or Local Coordinate System can be deleted from the model.

Other topological entities can also be deleted in certain cases.

- A vertex can be removed if the edges that link to it all have the same underlying geometric curve, e.g. represent adjoining parts of an arc of a circle.
- An edge can be removed if all faces touched by that edge have the same underlying surface geometry.
- Internal or external sheet faces can be deleted from a body. A sheet face is one that exists but supports no volume e.g. the product of copying a single face from a body creates such a face. An option to regularise allows any edges that are not needed to maintain the topology of the body to be removed from the body.
- External faces can be removed if the switch to allow this is activated. The removal will only occur if the resulting hole formed in the body can be patched. This cannot occur with faces that form part of a non-manifold geometry.
- Deleting a cell will remove all external faces that exist to support that cell. Any internal faces will now be external. An option to regularise allows removal of any edges and vertices that are no longer needed to maintain the topology of the body.

### *Checking And Correcting Topology*

In some instances, particularly after some more complex operations, or with data from other applications, the geometry and topology of the structure will be incorrect. A **CHECK** command exists to test for potential problems. Any entity that fails the check command will be flagged with a **SYSTEM** label, and the body to which it belongs will also be flagged. These bodies can be displayed and a warning message will give an indication of why the problem occurred. It may be necessary to remove and recreate any entities that exhibit problems, as these may cause future operations to fail.

## **Modeller Data**

Within the Modeller it is necessary to set properties defining material properties, mesh control information, volume properties and boundary conditions.

This information is used when creating the database for analysis by one of the analysis modules. It is also used to assist in visualisation of the model, as different data can be chosen as the basis for the display.

This data is stored with each entity with which it is associated, and is different for different types of topological entity.

### *Body Data*

A body can have:

- Name  
This is used for specifically naming bodies.
- Additional labels  
Used for grouping sections of the model for visualisation and modification.

When merging bodies the data from the body that is picked first is always maintained.

### *Cell Data*

A cell can have the following data attached to it:

- Material label  
A label that is used to reference the material properties associated with the cell. By default cells are labelled **AIR**, but other labels can be applied, and the properties e.g. permeability or BH curve, conductivity, permittivity, can be defined for each material label.
- Potential type (**TOTAL, REDUCED, VECTOR**)
- Element type (**LINEAR, QUADRATIC**)
- Volume properties label  
A volume property label can be attached to cells and stores other data associated with the cell. This can include velocity, local orientation, packing factor etc.
- Mesh control size, surface normal and normal distance tolerance  
Used to control the maximum size of mesh that is generated within this cell.
- Additional labels  
Used for grouping sections of the model for visualisation and modification.
- A data storage level  
When merging bodies with boolean operations, the resulting cells will normally be formed from a combination of the initial cells. In such a case it is not clear which of the initial data sets should be kept with each newly formed cell.

The data storage level is used to resolve this conflict, and the data set with the highest storage level is kept. Where the data level is the same, the result is ambiguous and the data is formed by merging the possible data sets.

### *Face Data*

A face can have the following data attached to it:

- Boundary condition label  
This is a label that is used to reference the boundary condition data for this face.
- Element type (**LINEAR**, **QUADRATIC**)
- Mesh control size, surface normal and normal distance tolerance  
Used to control the maximum size of element in the finite element mesh that is generated on this face.
- Additional labels  
Used for grouping sections of the model for visualisation and modification.
- A data storage level  
As with cells, merging may produce ambiguity in face properties. The data set with the greater storage level will be kept.

### *Edge And Vertex Data*

A face can have the following data attached to it:

- Mesh control size  
Used to control the maximum size of element in the finite element mesh that is generated on this face.
- Additional labels  
Used for grouping sections of the model for visualisation and modification.
- A data storage level  
As with cells, merging may produce ambiguity in edge properties. The data set with the greater storage level will be kept.

### *Setting Data Properties*

The different commands for setting properties are

<b>Command</b>	<b>Description</b>
<b>LABEL</b>	Adds and removes labels for all picked objects
<b>RENAME</b>	Gives names to and renames bodies and Local Coordinate Systems
<b>CELLDATA</b>	Sets the properties of picked cells

Command	Description ( <i>continued</i> )
<b>FACEDATA</b>	Sets the properties of picked faces
<b>EDGEDATA</b>	Sets properties of picked edges
<b>VERTEXDATA</b>	Sets properties of picked vertices
<b>MATERIALS</b>	Defines the data associated with a material label
<b>VOLUME</b>	Defines the data for each volume label
<b>BOUNDARY</b>	Defines the boundary condition data for each boundary label

Properties of one or more entities can be set together. For cells, faces, edges and vertices: the entities to be modified are picked. Any common data is displayed. This can be changed and the data set. If a data item is left blank, the original data is left unchanged.

### *Listing Data*

A **LIST** command can be used to list data associated with the picked entity. The **FILTER** command can also be used to adjust the effect of double clicking over a part of the model, so that the **LIST** command is automatically called instead of picking the item for modification.

## Display

The display uses an OpenGL interface to allow visualisation of the model. The model can be rotated, translated or zoomed using the mouse controls and the buttons in the display window.

### *View Selection*

The display of the model is controlled by the **SELECT** command. This allows individual items to be selected for display or to be explicitly hidden from view. There are many data items attached to objects, and these can be used to control the display of the model. The **SELECT** command allows the user to display or hide any object that has data of a certain type, e.g. any face with any boundary label, or more explicitly e.g. any face with a particular boundary label.

### *Contour Display*

Some data attributes can be contoured on surfaces of the model. The contours will only be shown on parts of the model that are visible through the normal display selection. The **CONTOUR** command gives greater detail.

### *Vector Display*

Some data attributes can be displayed as vectors within the model. The vectors will only be shown on parts of the model that are visible through the normal display selection. The **VECTOR** command gives greater detail.

## Command History

The Modeller maintains a history of model changes. Most commands will generate an entry within the history stream. The **UNDO** command is available to return to any previously noted state of the Modeller. After an **UNDO** command, a **REDO** command can undo the changes of the **UNDO** command, until a new command has been issued. At this point the **REDO** command becomes invalid, and the history between the current position and the position at which the **UNDO** command was issued is removed.

Certain commands such as **THREED**, **FILTER** and **SAVE** do not affect the model and do not create an entry in the history stream.

Other commands, such as the **SELECT** command, generate changes. However, when the command is issued several times in succession, the effects are merged into a single state within the history stream.

## Model Storage

The data within the Modeller can be saved at any time.

Once saved a file can be opened using the **LOAD** command. This will clear any existing data from Modeller and will overwrite it with data from the new file. The file can then be saved after modifications. A complete model, including the finite element mesh can be saved in a binary file.

If no file was opened a name must be given the first time the model is to be saved.

Data from a file can also be inserted into a currently open model. This will add the topology and data from the file into the open model.

Individual component bodies can be picked and exported to data file.

The standard data formats are the *.opc* file containing the OPERA-3d Modeller components and *.opcb* file which include the finite element mesh data. ACIS *.sat* data files containing geometry and data from other applications may be used within the OPERA-3d Modeller. It should be noted that in some cases, such data are not suitable for use with finite element analysis and may cause problems.

## Building a Model for Mesh Generation

To generate a continuous mesh through the volume of space modelled it is necessary to have a single body. This body is made up of multiple cells containing different material data representing the different components of the structure being modelled.

With such a potentially complex single body, it is difficult to modify the geometry and topology of the structure being modelled.

To assist in making modelling easier, it is preferable to have many simple bodies, possibly overlapping if necessary, that can be easily moved, scaled etc. as required.

To make the transition from multiple bodies to a single body suitable for meshing, there is a **MODEL** command. This command makes a copy of each body within the model and merges them using a boolean union operation, without regularisation to ensure that internal material boundaries are maintained. The single body formed will have the properties defined on the components, adjusted where necessary to take account of conflicts by using the data storage level. The model created should be suitable for meshing and database creation. If the model is not correct, it is possible to adjust the properties and topology, although these changes will be lost when returning to component view.

When creating the model, a **BACKGROUND** region can be used to limit the model space. This is useful for cases where the whole model is easy to define, e.g. cylinders, but where symmetry is available and so the whole model is not needed for analysis. Any body may represent the background region, and it is defined simply by having the name 'background'. The effect of the background region is that it is intersected with the final model to limit the extent of the model's size and shape. If boundary conditions have been attached, these will be maintained on the final model if the data storage level is set correctly.

Once created, if changes are again required, the **MODEL** can be deleted, and the view will return to the component view. If adjustments have been made to the model, these would be lost when returning to component view.

### *Mesh Generation*

Mesh generation is a two stage process. The **MESH** command generates a triangular surface mesh over all faces of the model. The mesh size is controlled by the mesh size, the surface, and the normal distance tolerances. The smallest of each of these is used to control the local mesh size. This command can only be used when the model has been created. The mesh can be displayed as outlines on the model display.

The **FILL** command creates the volume tetrahedral elements throughout the model space, and can only be used once a valid surface mesh exists.

Changes to the model topology will destroy the surface mesh, as will returning to component view. It is possible to **UNDO** back to the last meshes formed, but meshes prior to this will have been overwritten.

### *Database Generation*

General settings for analysis control are entered using the **ANALYSISDATA** command. The options used are dependent on the analysis type.

Control of the drive data for ELEKTRA-TR is set using the **DRIVE** command.

Output times for ELEKTRA-TR and frequencies for ELEKTRA-SS and SOPRANO-SS are set using the **DBCASEDATA** command.

When the model is ready with a volume mesh created, the database can be prepared using the **SOLVERS** command.

## Summary of Commands

---

### Commands

#### *Primitive Construction Commands*

Command	Description
<b>BLOCK</b>	Creates a new cuboid body
<b>CYLINDER</b>	Creates a new cylinder / cone body
<b>PRISM</b>	Creates an n-sided prism or pyramid body
<b>SPHERE</b>	Creates a new spherical body
<b>TORUS</b>	Creates a new toroid

#### *File And General Commands*

Command	Description
<b>CLEAR</b>	Clears and resets all data from the model
<b>END</b>	Ends this session of the Modeller
<b>HISTORY</b>	Controls the size of the history stream
<b>LOAD</b>	Loads model data from a file
<b>REDO</b>	Returns changes that have been undone
<b>SAVE</b>	Saves model data to a file
<b>UNDO</b>	Returns the model to a previous state

#### *Display And Selection Commands*

Command	Description
<b>COLOUR</b>	Sets the colour for display of items
<b>CONTOUR</b>	Displays contours of components on displayed entities
<b>FILTER</b>	Sets a filter and command for graphical selection
<b>HIDE</b>	Hides entities from display
<b>MOUSE</b>	Changes the operation of the right mouse button
<b>PICK</b>	Picks specific model entities for modification
<b>SELECT</b>	Selects the criteria which determine what is included in the display

<b>Command</b>	<b>Description (<i>continued</i>)</b>
<b>THREED</b>	Creates a display window for model visualisation
<b>TITLE</b>	Adds date, time and title to the display
<b>VECTOR</b>	Display vectors: material properties, boundary conditions or current directions
<b>WINDOW</b>	Controls display of parts of the 3d view

### *Conductor Command*

<b>Command</b>	<b>Description</b>
<b>ARC</b>	Creates or modifies an arc conductor
<b>BEDSTEAD</b>	Creates or modifies a bedstead conductor
<b>BRICK8</b>	Creates or modifies an 8-node brick conductor
<b>BRICK20</b>	Creates or modifies a 20-node brick conductor
<b>FITTEDCPE</b>	Creates or modifies a fitted constant perimeter end conductor
<b>HELICALEND</b>	Creates or modifies a helical end conductor
<b>RACETRACK</b>	Creates or modifies a racetrack conductor
<b>SOLENOID</b>	Creates or modifies a solenoid conductor
<b>STRAIGHT</b>	Creates or modifies a straight bar conductor
<b>TANGENTIALCPE</b>	Creates or modifies a tangential constant perimeter end conductor

### *Local Coordinate System Commands*

<b>Command</b>	<b>Description</b>
<b>LCS</b>	Creates a new Local Coordinate System (LCS)
<b>WCS</b>	Sets an Local Coordinate System as the Working Coordinate System (WCS)

### *Topology / Geometry Modification*

<b>Command</b>	<b>Description</b>
<b>BLEND</b>	Adds blends or chamfers at edges
<b>CHECK</b>	Checks the topology of parts of the model
<b>DELETE</b>	Deletes a body or parts of a body
<b>SWEEP</b>	Sweeps a face to form a volume

<b>Command</b>	<b>Description (<i>continued</i>)</b>
<b>TRANSFORM</b>	Transforms or copies existing bodies, conductors and Local Coordinate Systems
<b>COMBINE</b>	Combine (union, subtract, etc.) two or more bodies
<b>PRECISIONDATA</b>	Set geometric tolerances

### *Data Commands*

<b>Command</b>	<b>Description</b>
<b>BOUNDARY</b>	Sets the boundary conditions for a boundary label
<b>CELLDATA</b>	Sets the properties associated with cells
<b>EDGEDATA</b>	Sets the properties associated with edges
<b>FACEDATA</b>	Sets the properties associated with faces
<b>LABEL</b>	Attaches a label to any item
<b>LIST</b>	Lists data associated with any item
<b>MATERIALS</b>	Sets the material properties for a material label
<b>RENAME</b>	Renames a body or Local Coordinate System
<b>VERTEXDATA</b>	Sets the properties associated with vertices
<b>VOLUME</b>	Sets the properties for a volume label

### *Analysis Database Preparation Commands*

<b>Command</b>	<b>Description</b>
<b>ANALYSISDATA</b>	Sets analysis options used by the analysis modules
<b>DBCASEDATA</b>	Sets simulation output frequencies and times
<b>DRIVE</b>	Sets drive information for sources for ELEKTRA
<b>FILL</b>	Creates a volume mesh
<b>MESH</b>	Creates a surface mesh
<b>MODEL</b>	Creates (and deletes) a model suitable for analysis, from the components
<b>PERIODICITY</b>	Allows model periodicity to be used
<b>SOLVERS</b>	Creates a database for analysis
<b>VARIABLE</b>	Sets and stores user defined variables

## The **ANALYSISDATA** Command

### Summary

Sets the options for the analysis modules.

### Menu Route:

Model↓  
Analysis settings

### Command Line Parameters:

Command	ANALYSISDATA		
Parameter	Default	Function	
OPTION	none	LOAD	
		SET	
PROGRAM	none	Analysis program type	
		TOSCAMAGN	TOSCA: magnetic, electrostatic or current flow
		TOSCAELEC	
		TOSCACURR	
		SCALA	SCALA space charge beam analysis
		ELEKTRASS	ELEKTRA: Steady state harmonic, transient or velocity
		ELEKTRATR	
		ELEKTRAVL	
		SOPRANOSS	SOPRANO: Steady state harmonic or eigenvalue
SOPRANOEV			
LINEAR	YES	YES	Linear permeability for analysis
		NO	Non-linear permeability (TOSCAMAGN and ELEKTRA)
NLITERTYPE	NEWTON	NEWTON	Use Newton-Raphson non-linear updates
		SIMPLE	Use simple updates
NITERATIONS	21	Maximum number of non-linear iterations	
TOLERANCE	0.001	Non-linear iteration convergence tolerance	

Command	ANALYSISDATA (continued)		
Parameter	Default	Function	
RELAXATION	1.0	SCALA relaxation factor	
RHS	ADAPTIVE	RHS coil calculation method	
		SIMPLE	Simple coil integration
		ADAPTIVE	Adaptive coil integration
HXEXT	0	Externally applied H field for TOSCA-MAGN and SCALA	
HYEXT	0		
HZEXT	0		
UPWINDING	YES	NO	No upwinding for ELEKTRA-VL analysis
		YES	Use upwinding
ITPTSTEP	21	Non-linear iterations per time step (ELEKTRA-TR)	
UPDATE	SIMPLE	ELEKTRA-TR	time update method
		SIMPLE	Simple theta method fixed step updates
		AD2RK	Adaptive 2 <sup>nd</sup> order Runge-Kutta update
		AD4RK	Adaptive 4 <sup>th</sup> order Runge-Kutta update
		F2RK	Fixed 2 <sup>nd</sup> order Runge-Kutta update
		F4RK	Fixed 4 <sup>th</sup> order Runge-Kutta update
MAXADERR	1.0	Maximum error for adaptive updates	
DELTAT	0.01	Initial or fixed time step	
RPM	3000	Unused	
FREQ1	1.0e9	Lower range of SOPRANO-EV frequencies	
FREQ2	2.0e9	Upper range of SOPRANO-EV frequencies	
NEV	1	Number of eigenvalues to be found	
CONVTOL	1.0e8	Convergence tolerance for linear equation solutions	
PERIODICITY	YES	YES	Use periodicity
		NO	Do not use periodicity

Command	ANALYSISDATA (continued)		
Parameter	Default	Function	
POTENTIALCUT	YES	YES	Use automatic potential cuts
		NO	Do not use automatic potential cuts

### Notes

This command sets the options for the different analysis modules. Each module has its own independent set of data which is configured by this command.

Using **ANALYSISDATA OPTION=LOAD PROGRAM=TOSCAMAGN** will load the currently set options for the **TOSCA** magnetic analyses into the parameters. These can be set using **OPTION=SET** with new values for the parameters.

The **ANALYSISDATA** command sets various analysis options. From the menus, only those options relevant to the current simulation are available.

- **Linear or Non-linear:** Linear analyses use constant material properties; non-linear analyses update the material properties, depending on the solution and re-solve. Non-linear analyses can include some linear materials and linear analyses can include non-linear materials. In this latter case, the material property characteristic (e.g. BH curve) is used to supply the value of the material property (permeability). The slope of the curve at zero flux-density is used.
- **RHS Integrals:** The line and surface integrals of coil fields which are part of TOSCA magnetostatics and ELEKTRA analyses can be done using simple integrals, which only use the values of coil fields at the nodes or using adaptive integrals which use additional field points in order to reduce the errors.
- **External Fields** can be added to TOSCA magnetostatics analyses.
- **Automatic Potential cuts** can be used in TOSCA magnetostatic analyses to automatically insert potential cuts to avoid having multiply connected volumes, where a loop of total potential volume encloses a non-zero net current.
- **Periodicity** can be used in TOSCA and SCALA to avoid the need to build the full model where it is known that both field and geometry have rotational or translational symmetry.
- **SCALA Iterations** converge to a consistent set of particle trajectories and fields. The number of iterations and under-relaxation factor can be set.
- **Upwinding** is a technique to improve analysis of moving systems (ELEKTRA-VL). The analysis program reports whether upwinding is required or not.

- **Time stepping** in ELEKTRA-TR can use fixed time steps or can adjust the time step to achieve a given accuracy. If several similar models are to be analysed, the most efficient approach might be to use adaptive 4th order Runge-Kutta for one model to ascertain the appropriate time step to use. Subsequent models can then use a fixed time step method.

Each analysis program uses only a subset of the analysis parameters.

## The **ARC** Command

---

### Summary

Create or modify arc conductors.

### Icon



### Menu Route:

Create↓  
     Conductor → Arc  
 Operations↓  
     Modify conductors → Arc

### Command Line Parameters:

Command	ARC		
Parameter	Default	Function	
OPTION	NEW	NEW	Creates a new arc conductor
		MODIFY	Modifies properties of the picked arc conductors
		LOAD	Loads defaults from picked conductors
DRIVELABEL		Name for the arc drive label	
LCNAME		Name for Local Coordinate System for coordinate system 1	
SYMMETRY		Rotational symmetry about global Z axis	
X0		Origin of coordinate system 2	
Y0			
Z0			
THETA		Euler angles defining orientation of coordinate system 2	
PHI			
PSI			
RXY		Reflection symmetries in XY, YZ and ZX planes.	
RYZ			
RZX			
A		Cross-sectional width	

Command	ARC (continued)		
Parameter	Default	Function	
B		Cross-sectional height	
R1		Radius of the arc	
ANGLE		Angle subtended by the arc	
CURD		Current density in the conductor	
TOLERANCE		Field calculation tolerance	
KEEP	NO	NO	Clear the list of picked items
		YES	Keep the list of picked items for further modification

### Notes

This command creates a new arc conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked arcs, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked arc conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked arcs. If the picked arcs do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked arcs to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see “Circular Arcs” on page 4-36.

## The **BEDSTEAD** Command

---

**Summary** Create or modify bedstead conductors.

**Icon**



**Menu Route:**

Create↓  
     Conductor → Bedstead  
 Operations↓  
     Modify conductors → Bedstead

**Command Line Parameters:**

Command	BEDSTEAD		
Parameter	Default	Function	
OPTION	NEW	NEW	Creates a new bedstead conductor
		MODIFY	Modifies properties of the picked bedstead conductors
		LOAD	Loads defaults from picked conductors
DRIVELABEL		Name for the bedstead drive label	
LCNAME		Name for Local Coordinate System for coordinate system 1	
SYMMETRY		Rotational symmetry about global Z axis	
X0		Origin of coordinate system 2	
Y0			
Z0			
THETA		Euler angles defining orientation of coordinate system 2	
PHI			
PSI			
RXY		Reflection symmetries in XY, YZ and ZX planes	
RYZ			
RZX			

Command	BEDSTEAD (continued)		
Parameter	Default	Function	
XP1		Local coordinates of lower, inside corner	
YP1			
A		Cross-sectional width	
B		Cross-sectional height	
H1		Half-length of the straight	
H2		Half length of the upright	
R1		Radius of the arc	
R2		Radius of the upper bend	
CURD		Current density in the conductor	
TOLERANCE		Field calculation tolerance	
KEEP	NO	NO	Clear the list of picked items
		YES	Keep the list of picked items for further modification

### Notes

This command creates a new bedstead conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked bedsteads, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked bedstead conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked bedsteads. If the picked bedsteads do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked bedsteads to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see “Bedsteads” on page 4-30.

## The **BLEND** Command

---

### Summary

Creates a blend or chamfer at edges between adjoining faces.

### Menu Route:

Operations↓  
Blend or chamfer edges

### Command Line Parameters:

Command	<b>BLEND</b>		
Parameter	Default	Function	
<b>OPTION</b>	<b>BLEND</b>	<b>BLEND</b>	Create a blend
		<b>CHAMFER</b>	Create a chamfer
<b>RADIUS</b>		Radius of the blend	
<b>LEFTCHAMFER</b>		Distance to chamfer from the edge on the first face	
<b>RIGHTCHAMFER</b>		Distance to chamfer from the edge on the adjoining face	

### Notes

This command operates on a set of picked edges. The edges must be non-manifold (i.e. there must not be more than two faces meeting at the edge).

If **OPTION=BLEND**, then for each picked edge a tangential join is formed. The join is cylindrical in nature, with radius given by the parameter **RADIUS**.

If **OPTION=CHAMFER**, the edge is planed off to give a planar face join between the two faces, with each of the original faces cut by the **LEFTCHAMFER** or **RIGHTCHAMFER** distances. At present there is no way to distinguish which value is applied to which face, so the operation must be **UNDONE** and re-applied with the values switched if they are incorrect at the first attempt.

When applying the command to multiple edges, the result may differ from the result of applying the command to each edge individually. The **BLEND** command may fail if no suitable blend or chamfer surface can be found.

## The **BLOCK** Command

---

**Summary**            Creates a cuboid block from the data supplied.

**Icon**



**Menu Route:**        Create↓  
                          Object → Block

**Command Line  
Parameters:**

Command	<b>BLOCK</b>	
Parameter	Default	Function
<b>NAME</b>		Attaches this name to the cuboid body formed
<b>X0</b>		Coordinates of corner of the cuboid
<b>Y0</b>		
<b>Z0</b>		
<b>X1</b>		Coordinates of opposite point in the cuboid
<b>Y1</b>		
<b>Z1</b>		

**Notes**                The body formed must have non-zero area in the XY plane, i.e. **X0** is not equal to **X1**, **Y0** is not equal to **Y1**. If **Z0** equals **Z1** a planar body of zero volume will be created.

The coordinates specified are in the Working Coordinate System.

## The **BOUNDARY** Command

---

**Summary** Sets the properties associated with a boundary label.

**Menu Route:** Model↓  
Set boundary conditions

**Command Line Parameters:**

Command	<b>BOUNDARY</b>		
Parameter	Default	Function	
<b>OPTION</b>		<b>PICK</b>	Adds a boundary label to a list to be set
		<b>UNPICK</b>	Clears the list of picked boundary labels
		<b>RESET</b>	Clears the data from the picked boundary labels
		<b>MODIFY</b>	Sets the data for the picked boundary labels
		<b>LIST</b>	Lists the boundary conditions of picked labels
		<b>DELETE</b>	Deletes the picked labels
<b>BOUNDARYLABEL</b>		Boundary label to be picked	
<b>CONDITION</b>		<b>NONE</b>	None
		<b>TANGMAGN</b>	Tangential magnetic
		<b>NORMMAGN</b>	Normal magnetic
		<b>NORMMAGP</b>	Normal magnetic with assigned constant potential
		<b>TANGELEC</b>	Tangential electric
		<b>NORMELEC</b>	Normal electric
		<b>NORMELEV</b>	Normal electric with assigned constant voltage
		<b>POTENTIAL</b>	Functional magnetic scalar potential
		<b>VOLTAGE</b>	Functional voltage

Command	BOUNDARY (continued)	
Parameter	Default	Function
		<b>MIXED</b> Mixed potential condition
		<b>DPDN</b> Assigned derivative of scalar potential
		<b>DVDN</b> Assigned derivative of voltage
		<b>VECTOR</b> Assigned vector potential
		<b>IVECTOR</b> Assigned incident vector potential
		<b>RADIATION</b> Radiation
		<b>PEC</b> Perfect conductor
		<b>SYMMETRY</b> Symmetry boundary
<b>VOLTAGE</b>		Functional electric scalar potential
<b>DVOLTAGE</b>		Derivative of electric scalar potential
<b>MPOTENTIAL</b>		Functional magnetic scalar potential
<b>DMPOTENTIAL</b>		Derivative of magnetic scalar potential
<b>AX</b>		Components of magnetic vector potential
<b>AY</b>		
<b>AZ</b>		
<b>INAX</b>		Components of incident magnetic vector potential
<b>INAY</b>		
<b>INAZ</b>		
<b>PMIXA</b>		Mixed derivative and potential condition coefficients
<b>PMIXB</b>		
<b>CMPOTENTIAL</b>		Value for constant magnetic scalar potential
<b>CVOLTAGE</b>		Value for constant voltage
<b>DRIVELABEL</b>		Drivelabel for functional time variation of assigned values

### Notes

This command defines the boundary conditions for use by the analysis programs.

A set of boundary labels is picked using the command repeatedly, with **OPTION=PICK** and a **BOUNDARYLABEL** specified. A boundary label can be removed from the set using **OPTION=UNPICK**. If no **BOUNDARYLABEL** is given, the set is emptied.

Issuing the command with **OPTION=MODIFY** will modify the properties of the set of picked boundary labels to the new values given in the parameters. The value of a property associated with the boundary labels is unchanged if the parameter is unset.

The default value of each of the parameters is updated to be the common value of all the picked boundary labels. If the data of one of these parameters is unset, or the picked boundary labels do not share the same value, then the parameter value is left clear.

**OPTION=RESET** will clear the properties associated with all of the picked boundary labels.

The properties of all boundary labels can be listed using **OPTION=LIST**.

Boundary labels that are not used, i.e. have no face referencing them can be deleted using **OPTION=DELETE**. Deleting a boundary label that is in use will clear its properties.

All parameters can be specified. The value of **CONDITION** determines which will be used.

The **DRIVELABEL** allows value and functional based boundary conditions to be driven in ELEKTRA-TR or assigned a phase lag in ELEKTRA-SS and SOPRANO-SS.

## The **BRICK8** Command

---

**Summary** Create or modify 8-node bricks.

**Icon**



**Menu Route:**

Create↓  
     Conductor → 8-node brick  
 Operations↓  
     Modify conductors → 20-node brick

**Command Line Parameters:**

Command	BRICK8		
Parameter	Default	Function	
OPTION	NEW	NEW	Creates a new 8-node brick conductor
		MODIFY	Modifies properties of the picked 8-node brick conductors
		LOAD	Loads defaults from picked conductors
DRIVELABEL		Name for the drive label	
LCNAME		Name for Local Coordinate System for coordinate system 1	
SYMMETRY		Rotational symmetry about global Z axis	
X0		Origin of coordinate system 2	
Y0			
Z0			
THETA		Euler angles defining orientation of coordinate system 2	
PHI			
PSI			
RXY		Reflection symmetries in XY, YZ and ZX planes	
RYZ			
RZX			

Command	<b>BRICK8</b> ( <i>continued</i> )		
Parameter	Default	Function	
<b>XP1</b>		First point defining the conductor corners	
<b>YP1</b>			
<b>ZP1</b>			
...			
<b>XP8</b>		Last point defining the conductor corners	
<b>YP8</b>			
<b>ZP8</b>			
<b>CURD</b>		Current density in the conductor	
<b>TOLERANCE</b>		Field calculation tolerance	
<b>KEEP</b>	<b>NO</b>	<b>NO</b>	Clear the list of picked items
		<b>YES</b>	Keep the list of picked items for further modification

### Notes

This command creates a new 8-node brick conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked 8-node bricks, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked 8-node brick conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked 8-node brick conductors. If the picked 8-node bricks do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked 8-node bricks to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see [“Bricks” on page 4-37](#).

## The **BRICK20** Command

---

**Summary** Create or modify 20-node bricks.

**icon**



**Menu Route:**

Create↓  
     Conductor → 8-node brick  
 Operations↓  
     Modify conductors → 20-node brick

**Command Line Parameters:**

Command	BRICK20		
Parameter	Default	Function	
OPTION	NEW	NEW	Creates a new 8-node brick conductor
		MODIFY	Modifies properties of the picked 8-node brick conductors
		LOAD	Loads defaults from picked conductors
DRIVELABEL		Name for the drive label	
LCNAME		Name for Local Coordinate System for coordinate system 1	
SYMMETRY		Rotational symmetry about global Z axis	
X0		Origin of coordinate system 2	
Y0			
Z0			
THETA		Euler angles defining orientation of coordinate system 2	
PHI			
PSI			
RXY		Reflection symmetries in XY, YZ and ZX planes	
RYZ			
RZX			

Command	<b>BRICK20</b> ( <i>continued</i> )		
Parameter	Default	Function	
<b>XP1</b>		First point defining the conductor corners	
<b>YP1</b>			
<b>ZP1</b>			
...			
<b>XP20</b>		Last point defining the conductor corners	
<b>YP20</b>			
<b>ZP20</b>			
<b>CURD</b>		Current density in the conductor	
<b>TOLERANCE</b>		Field calculation tolerance	
<b>KEEP</b>	<b>NO</b>	<b>NO</b>	Clear the list of picked items
		<b>YES</b>	Keep the list of picked items for further modification

### Notes

This command creates a new 20-node brick conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked 20-node bricks, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked 20-node brick conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked 20-node brick conductors. If the picked 20-node bricks do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked 20-node bricks to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see “Bricks” on page 4-37.

## The **CELLDATA** Command

---

**Summary** Sets properties of picked cells.

**Menu Route:** Properties↓  
Cell properties

**Command Line Parameters:**

Command	<b>CELLDATA</b>		
Parameter	Default	Function	
<b>OPTION</b>	<b>MODIFY</b>	<b>MODIFY</b>	Applies new values to the picked cells
		<b>RESET</b>	Clears all data and sets to the default values
<b>MATERIAL</b>	See notes	Material label	
<b>POTENTIAL</b>	See notes	<b>TOTAL</b>	Potential type of the volume
		<b>REDUCED</b>	
		<b>VECTOR</b>	
<b>ELEMTYPE</b>	See notes	<b>LINEAR</b>	Type of finite element in the volume
		<b>QUADRATIC</b>	
<b>VOLUME</b>	See notes	Volume property label	
<b>SIZE</b>	See notes	Mesh control size	
<b>NORMALTOL</b>	See notes	Maximum normal angle between mesh nodes	
<b>SURFACETOL</b>	See notes	Maximum deviation of the mesh from the surface	
<b>LEVEL</b>	See notes	Data storage level for the cell data	

**Notes** This command is used to set or clear the properties of all picked cells.

When a cell is created, it has default properties of

- **MATERIAL=**AIR,
- **POTENTIAL=**REDUCED
- **ELEMTYPE=**LINEAR.

These properties are also set when the cell's data is cleared using **OPTION=RESET**.

**OPTION=MODIFY** can be used to change the value of all of the picked cells. Any unset parameter values are not modified. The new value of parameters that have been set replace the existing values in the data attached to the cells.

Each time a cell is picked the value of each of the parameters is updated to be the common value of all the picked cells. If the data of one of these parameters is unset, or the picked cells do not share the same value, then the parameter value is left clear.

After issuing the command with **OPTION=MODIFY**, the list of picked items is cleared. Issuing the command with **OPTION=RESET** keeps the same set of picked objects.

The **MATERIAL** parameter controls the material label attached to a cell. The properties associated with such a label are set using the **MATERIALS** command.

The **ELEMTYPE** parameter allows the user to force the use of quadratic elements when creating the database. All elements in the cell will be quadratic if the database is created with mixed elements. See the **SOLVERS** command for greater detail.

The **VOLUME** parameter holds the volume property label that may be attached to a cell. The properties associated with such a label are set using the **VOLUME** command.

The **SIZE**, **NORMALTOL** and **SURFACETOL** parameters control the mesh size of elements in the cell and near the faces of the cell, when generating the surface and volume mesh.

The **LEVEL** parameter controls the storage of data when there is a conflict during the merging of multiple cells. The data set with the greater level will be maintained. The result of merging 2 cells with the same level is indeterminate.

## The **CHECK** Command

---

**Summary** Checks a picked body for topological problems.

**Menu Route:** Operations↓  
Check

**Command Line Parameters:**

Command	<b>CHECK</b>
No parameters	

**Notes** This command checks for topological problems of a single picked body. If problems exist, the objects are given a **SYSTEM** label and information regarding the problem is also attached to the body, or constituent parts of the body. This can be seen by listing the data on the object through the **FILTER COMMAND=LIST** command.

If problems exist with a body, it may be necessary to undo the operation that caused the problem and try an alternative method to generate the model. Failure to do this could cause the body to fail in future operations.

There are many reasons why a body fails the check command. The determination of this reason is currently very restricted.

## The **CLEAR** Command

---

**Summary**            Clears the model of all data, or reloads the open file.

**Menu Route:**        File↓  
                               Close  
                               File↓  
                               Revert to saved

**Command Line  
 Parameters:**

Command	<b>CLEAR</b>		
Parameter	Default	Function	
<b>REVERT</b>	<b>NO</b>	<b>NO</b>	Clears and re-initialises all data within the Modeller
		<b>YES</b>	Clears all data and reloads the previously open file

**Notes**                This command is used to reset the Modeller to its initial state. If **REVERT=YES**, the open file is re-loaded to remove changes from the model that are not required.

All history states are cleared, so this operation cannot be undone.

## The **COLOUR** Command

---

**Summary** Sets the colour attached to a display label

**Icon**



**Menu Route:** View ↓  
Change colours

**Command Line Parameters:**

Command	<b>COLOUR</b>	
Parameter	Default	Function
<b>CODE</b>		Code number for the colour to be altered
<b>PROPERTY</b>		Property type for item to be set
<b>LABEL</b>		Individual label of property
<b>RED</b>		Value of red component (0 - 255)
<b>GREEN</b>		Value of green component (0 - 255)
<b>BLUE</b>		Value of blue component (0 - 255)

**Notes** This command sets the **COLOUR**s used for the display of the model.

Properties displayed within the Modeller are assigned an integer colour **CODE**. If this code number has been specified as a parameter this will be used. If not, a property and label can be specified, and the code for this property will be found and used, e.g. **PROPERTY=MATERIAL** and **LABEL=AIR** will change the colour of the display of items with material label air.

The colour is changed to the values of **RED**, **GREEN** and **BLUE** supplied. If any value is not given, this component remains unchanged.

## The **COMBINE** Command

---

**Summary** Combines picked bodies using a boolean operation.

**Menu Route:** Operations↓  
Combine picked bodies

**Command Line Parameters:**

Command	<b>COMBINE</b>		
Parameter	Default	Function	
<b>OPTION</b>	<i>none</i>	<b>UNION</b>	Forms the union of the picked bodies
		<b>INTERSECT</b>	Returns the intersection of the bodies
		<b>SUBTRACT</b>	Subtracts all other picked bodies from the first body picked
		<b>TRIM</b>	Subtracts images of other picked bodies from the first body, leaving these intact
		<b>CUTAWAY</b>	Leaves the first picked body unaffected, but subtracts it from all other picked bodies
<b>REGULAR</b>	<b>YES</b>	<b>YES</b>	Regularise the result of the boolean operation
		<b>NO</b>	Do not regularise the result

**Notes** This command combines all picked bodies using the specified operation. At least 2 bodies must be picked for these operations.

**UNION** will form the sum of the picked bodies in a single body. After the operation the original bodies no longer exist within the model.

**INTERSECT** keeps the common volume that exists within all picked bodies. After the operation the original bodies no longer exist within the model.

**SUBTRACT** returns a body containing cells occupying the volume that exists in the first body picked, and which does not exist in any of the subsequently picked bodies. After the operation the original bodies no longer exist within the model.

**TRIM** leaves all picked bodies except the first untouched, and trims the first body so that it has no overlap with the other picked bodies.

**CUTAWAY** leaves only the first picked body untouched, and cuts away the first body from each of the others, so that it has no overlap with the other picked bodies.

If **REGULAR=YES**, faces, edges and vertices that are affected by the boolean operation are checked to see if they are needed to support the geometry. If not, they are removed and the body simplified. Internal faces formed by the operation are not needed to support the geometry, and so are removed. Faces bounding cells that have no volume, i.e. sheets, are also removed. Edges that bound two faces with the same underlying geometry are removed.

If the bodies do not overlap, the **INTERSECT** operation will have the effect of removing all the picked bodies.

If the first picked body is completely surrounded by the other picked bodies, the **SUBTRACT** operation will also remove all picked bodies.

It should be noted that the non-regular operations may leave sheet faces. Such sheet faces can be picked and deleted using the **DELETE** command ([page 3-49](#)).

## The **COMMENT** Command

---

**Summary** Adds comments into the database when creating a new simulation for analysis.

**Command Line Parameters:**

Command	<b>COMMENT</b>		
Parameter	Default	Function	
<b>TEXT</b>	<i>none</i>	Line of text to be added to the stored comment text	
<b>CLEAR</b>	<b>NO</b>	<b>NO</b>	Add additional text to the stored comments
		<b>YES</b>	Clears existing stored text
<b>TYPE</b>	<b>DBTITLE</b>	<b>DBTITLE</b>	Type of comment

**Notes** This command adds text as a comment for storing with a database simulation. These comments can be used to add information about the simulation for later reference.

Lines of text are added into a stored string. This stored string can be cleared using **CLEAR=YES**.

## The **CONDUCTOR** Command

---

### Summary

Modifies data common to all picked conductors of any type.

### Menu Route:

Operations↓

Modify conductors → Any conductor type

### Command Line Parameters:

Command	<b>CONDUCTOR</b>		
Parameter	Default	Function	
<b>OPTION</b>	<i>none</i>	<b>MODIFY</b>	Modifies properties of the picked conductors
		<b>LOAD</b>	Loads defaults from picked conductors
<b>DRIVELABEL</b>		Name for the conductor drive label	
<b>LCNAME</b>		Name for Local Coordinate System for coordinate system 1	
<b>SYMMETRY</b>		Rotational symmetry about global Z axis	
<b>X0</b>		Origin of coordinate system 2	
<b>Y0</b>			
<b>Z0</b>			
<b>THETA</b>		Euler angles defining orientation of coordinate system 2	
<b>PHI</b>			
<b>PSI</b>			
<b>RXY</b>		Reflection symmetries in XY, YZ and ZX planes.	
<b>RYZ</b>			
<b>RZX</b>			
<b>CURD</b>		Current density in the conductor	
<b>TOLERANCE</b>		Field calculation tolerance	
<b>KEEP</b>	<b>NO</b>	<b>NO</b>	Clear the list of picked items
		<b>YES</b>	Keep the list of picked items for further modification

**Notes**

This command operates on the list of picked conductors of any type, and can be used to change common properties such as coordinate systems, symmetries, current density, tolerance etc.

This command cannot be used to change the parameters defining the geometry of the conductors. To modify the geometry of e.g. a solenoid, the **SOLENOID** command should be used.

The default values of the command parameters are updated to match common values shared by all of the picked conductors. If the picked conductors do not have a common value for a parameter, that parameter is left unset. **CONDUCTOR MODIFY** will change the conductor data of all of the picked conductors to the new values given in the parameters, but will not affect the conductor data of any unset parameter. The current common values of the picked conductors are available by calling **CONDUCTOR OPTION=LOAD**.

For more information on the parameters, see “[The CONDUCTOR Sub-command MODIFY](#)” on page 4-44.

## The **CONTOUR** Command

---

**Summary** Displays contour of attached data on the model.

**Menu Route:** View↓  
Contours

**Command Line Parameters:**

Command	<b>CONTOUR</b>		
Parameter	Default	Function	
<b>COMPONENT</b>	<b>NONE</b>	<b>NONE</b>	No contours will be displayed
		<b>VOLTAGE</b>	Voltage contour
		<b>MPOENTIAL</b>	Magnetic potential contours
		<b>CHARGE</b>	Charge density
		<b>ROTATION</b>	Rotational velocity
		<b>AX</b>	Magnetic vector potential
		<b>AY</b>	
		<b>AZ</b>	
		<b>AMOD</b>	
		<b>VX</b>	Applied linear velocity
		<b>VY</b>	
		<b>VZ</b>	
		<b>VMOD</b>	
		<b>JX</b>	Applied source current density
		<b>JY</b>	
		<b>JZ</b>	
<b>JMOD</b>			

**Notes** This command displays contours of the **COMPONENT** on surfaces of objects in the model. If an object has data containing the component, then the value of the data will be contoured over it. Other displayed objects will be displayed normally. The contour command options are based on data that can be set on cells and faces only.

If the **CONTOUR** command is used to display functional components, the picture on the screen depends on the existence of a surface mesh.

With the surface mesh generated, the display will be based on the mesh that has been formed, and it will show the distribution of the component which is passed on to the analysis module.

If the surface mesh is not formed yet, the display is based on the surface display facets, which may not represent the final mesh very well. In such a case, a large non-linearity in functional definitions of these values may not be seen over the surface.

## The **CYLINDER** Command

---

**Summary**            Creates a cylinder or cone, with either circular or elliptic base.

**Icon**



**Menu Route:**        Create↓  
                              Object → Cylinder / cone

**Command Line  
Parameters:**

Command	<b>CYLINDER</b>	
Parameter	Default	Function
<b>NAME</b>		Attaches this name to the body formed
<b>X0</b>		X coordinate of central point on the base
<b>Y0</b>		Y coordinate of central point on the base
<b>Z0</b>		Z coordinate of central point on the base
<b>X1</b>		X coordinate of central point on the top
<b>Y1</b>		Y coordinate of central point at the top
<b>Z1</b>		Z coordinate of central point at the top
<b>MAJORRADIUS</b>		Major radius of the base of the cylinder
<b>MINORRADIUS</b>		Minor radius of the base of the cylinder
<b>TOPRADIUS</b>		Radius at the top of the cylinder

**Notes**

All values must be given. The axis of the body is formed between points **(X0,Y0,Z0)** and **(X1,Y1,Z1)** and the normals of the base and top planes are parallel to this. The base point and top point must be different, and the radii given must all be greater than zero.

If the **TOPRADIUS** is given as zero, the body formed is a cone.

If the **MAJORRADIUS** is not the same as the **MINORRADIUS**, the body formed has an elliptic base.

The coordinates specified are in the Working Coordinate System.

## The **DBCASEDATA** Command

---

### Summary

Sets the list of simulations frequencies for ELEKTRA-SS and SOPRANO-SS and the list of output times for ELEKTRA-TR analyses.

### Menu Route:

Model↓  
Analysis settings

### Command Line Parameters:

Command	DBCASEDATA		
Parameter	Default	Function	
<b>PROGRAM</b>	<i>none</i>	<b>ELEKTRASS</b>	Set frequencies for <b>ELEKTRA-SS</b>
		<b>ELEKTRATR</b>	Set output times for <b>ELEKTRA-TR</b>
		<b>SOPRANOSS</b>	Set frequencies for <b>SOPRANO-SS</b>
<b>OPTION</b>	<b>ADD</b>	<b>ADD</b>	Add a new frequency / output time to the list
		<b>INSERT</b>	Insert a frequency at a position in the list
		<b>DELETE</b>	Delete an item from the list
		<b>REPLACE</b>	Replace an item in the list
		<b>LIST</b>	Lists the set of frequencies / output times.
<b>INDEX</b>	1	The position at which to insert, delete or replace	
<b>VALUE</b>		The new value for add, insert or replace	

### Notes

This command creates a list of frequencies or output times for the different analysis programs. Each program has a separate list of values.

- ELEKTRA-SS or SOPRANO-SS data bases will be analysed for each of the frequencies given in the list.
- An ELEKTRA-TR solution will be output at each of the times in the list. The output time list is automatically chronologically ordered, so that insert and add have the same effect. An output time at t=0 is also added.

All lists automatically remove duplicates.

## The **DELETE** Command

---

**Summary** Deletes the picked entities.

**Menu Route:** Operations↓  
Delete

**Command Line Parameters:**

Command	<b>DELETE</b>		
Parameter	Default	Function	
<b>REGULARISE</b>	<b>YES</b>	<b>YES</b>	Regularise the result
		<b>NO</b>	Do not regularise the result
<b>EXTERNAL</b>	<b>NO</b>	<b>YES</b>	Allow external faces to be deleted
		<b>NO</b>	Do not delete external faces

**Notes** This command tries to **DELETE** all picked objects, including Local Coordinate Systems and conductors.

If bodies have been picked, these will always be deleted. If cells have been picked, the delete command will remove the cells from the body and delete them. If a body only has a single cell, deleting the cell has the same effect as deleting the body.

If a face to be deleted is internal to a body, or if the face does not form part of a cell, i.e. both sides are external, the face can be deleted. If the face is internal, the two cells to either side of the face will be merged. If regularise is on, the edges and vertices that formed the boundary of the face are tested. If they are not required to support the geometry, they are deleted as well.

Faces forming the external boundaries of a body can be deleted only if the gap formed can be patched by the neighbouring faces. Therefore, one side of a cube could not be deleted because the neighbouring faces would be unable to patch it. The parameter **EXTERNAL=YES** must be used to allow these faces to be deleted.

Edges and vertices may be deleted if they are not required to support the topology of the body, e.g. if the face to both sides of an edge has the same underlying geometric surface.

## The **DRIVE** Command

---

**Summary** Sets the properties associated with a drive label.

**Menu Route:** Model↓  
Set drive properties

**Command Line Parameters:**

Command	DRIVE	
Parameter	Default	Function
<b>OPTION</b>		<b>PICK</b> Adds a drive label to a list to be set
		<b>UNPICK</b> Clears the list of picked drive labels
		<b>RESET</b> Clears the data from the picked drive labels
		<b>MODIFY</b> Sets the data for the picked drive labels
		<b>LIST</b> Lists the drive data of picked labels
		<b>DELETE</b> Deletes the picked labels
<b>DRIVELABEL</b>		Drive label to be picked
<b>TYPE</b>		<b>DC</b> Drive function associated with the label
		<b>STEP</b> Step at time zero
		<b>RAMP</b> Ramp drive
		<b>SINE</b> Sine drive
		<b>COSINE</b> Cos drive
		<b>PEAK</b> Rise to peak value before decaying
		<b>RISE</b> Exponential rise function
		<b>TABLE</b> Switch on table
	<b>TOFF</b> Switch off table	
<b>SSPHASE</b>		Phase lag used during ELEKTRA-SS and SOPRANO-SS analysis
<b>SINFREQUENCY</b>		Frequency for Sine drives

Command	DRIVE (continued)	
Parameter	Default	Function
SINPHASE		Phase for Sine drives
COSFREQUENCY		Frequency for Cosine drives
COSPHASE		Phase for Cosine drives
STEPTIME		Unused
RAMPTIME		Time taken to ramp
PEAKTIME		Time at which peak occurs
RISETIME		Time constant for the rise
TABLEFILE		File for switch on timetable information
TABLEOFFFILE		File for switch off timetable information
ROTATING		Not used at present

### Notes

This command defines the drive functions for use by the analysis programs, ELEKTRA-TR, ELEKTRA-SS. For ELEKTRA-SS all data except for the phase is ignored.

A set of drive labels is picked using the command repeatedly, with **OPTION=PICK** and a **DRIVELABEL** specified. A drive label can be removed from the set using **OPTION=UNPICK**. If no **DRIVELABEL** is given, the set is emptied.

Issuing the command with **OPTION=MODIFY** will modify the properties of the set of picked drive labels to the new values given in the parameters. The value of a property associated with the drive labels is unchanged if the parameter is unset.

The default value of each of the parameters is updated to be the common value of all the picked volume labels. If the data of one of these parameters is unset, or the picked drive labels do not share the same value, then the parameter value is left clear.

**OPTION=RESET** will clear the properties associated with all of the picked drive labels.

The properties of all drive labels can be listed using **OPTION=LIST**.

Drive labels that are not used, i.e. have no boundary condition or conductor referencing them can be deleted using **OPTION=DELETE**. Deleting a drive label that is in use will reset its properties.

ELEKTRA-SS and SOPRANO-SS only use the **SSPHASE** parameter information. If this is unset, it is assumed to be 0.

ELEKTRA-TR drives are mainly controlled by the **TYPE** parameter. Which of the other parameters are used is determined from the option assigned to type. If unset, a **TYPE=DC** is assumed.

## The **EDGEDATA** Command

---

**Summary** Sets properties of picked edges.

**Menu Route:** Properties↓  
Edge properties

**Command Line Parameters:**

Command	<b>EDGEDATA</b>		
Parameter	Default	Function	
<b>OPTION</b>	<b>MODIFY</b>	<b>MODIFY</b>	Applies new values to the picked edges
		<b>RESET</b>	Clears all data from the picked edges
<b>SIZE</b>	See notes	Mesh control size	
<b>LEVEL</b>	See notes	Data storage level for the edge data	

**Notes** This command is used to set or clear the properties of all picked edges.

Edges initially have no data assigned to them.

If issuing the command with **OPTION=MODIFY**, new values of parameters that have been set replace the existing values of data on the edges. The data on the edges, associated with any unset parameters, is left unchanged.

The default value of each of the parameters is updated to be the common value of all the picked edges. If the data of one of these parameters is unset, or the picked edges do not share the same value, then the parameter value is left clear.

**SIZE** is the maximum length of an element side along the edge.

The **LEVEL** parameter controls the storage of data when there is conflict during the merging of multiple edges. The data set with the greater level will be maintained. The result of merging 2 edges with the same level is indeterminate.

Upon issuing the command with **OPTION=MODIFY**, all picked items are deselected. Issuing the command with **OPTION=RESET** keeps the same set of picked objects.

## The **END** Command

---

**Summary**            Ends the current Modeller session.

**Menu Route:**        **File**↓  
                              **Exit**

**Command Line  
Parameters:**

Command	<b>CLEAR</b>	
Parameter	Default	Function
<i>none</i>		

**Notes**                This command ends the current session of the Modeller. If there have been changes since the last **SAVE OPTION=ALL** or **SAVE OPTION=NEW** command, then confirmation of the end command will be required.

## The **EXPORT** Command

---

**Summary** Exports conductor data to a file for use by the Modeller, pre, or post processor.

**Menu Route:** Create↓  
Conductor → Export

**Command Line Parameters:**

Command	<b>EXPORT</b>	
Parameter	Default	Function
<b>FILE</b>		Conductor file to be created

**Notes** The **EXPORT** command writes a file containing all conductors in the model. This file can be read by the pre processor **READ** command (page 4-149), the post processor **CONDUCTOR** command (page 5-39), and the Modeller **IMPORT** command (page 3-67).

The conductor data in the file contains the values for coordinate system 1, not the name of the Local Coordinate System with which the conductor is associated.

## The **FACEDATA** Command

---

**Summary**            Sets properties of picked faces.

**Menu Route:**        Properties↓  
                          Face properties

**Command Line  
Parameters:**

Command	<b>FACEDATA</b>		
Parameter	Default	Function	
<b>OPTION</b>	<b>MODIFY</b>	<b>MODIFY</b>	Applies new values to the picked faces
		<b>RESET</b>	Clears all data from the picked faces
<b>BOUNDARY</b>	See notes	Boundary label on the face	
<b>ELEMTYPE</b>	See notes	<b>LINEAR</b>	Elements touching the face will be linear
		<b>QUADRATIC</b>	Elements touching this face will be quadratic
<b>SIZE</b>	See notes	Mesh control size	
<b>NORMALTOL</b>	See notes	Maximum normal angle between mesh nodes	
<b>SURFACETOL</b>	See notes	Maximum deviation of the mesh from the surface	
<b>LEVEL</b>	See notes	Sets the data storage level for the face data	

**Notes**                This command is used to set or clear the properties of all picked faces.

Faces initially have no data assigned to them, and **OPTION=RESET** will clear all data of the picked faces.

Issuing the command with **OPTION=MODIFY** will modify the properties of the picked faces to the values given in the parameters. The value of properties on the picked faces are unchanged if the parameter is unset.

The default value of each of the parameters is updated to be the common value of all the picked faces. If the data of one of these parameters is unset, or the picked faces do not share the same value, then the parameter value is left clear.

Upon issuing the command with **OPTION=MODIFY**, the list of picked items is cleared. Issuing the command with **OPTION=RESET** keeps the same set of picked objects.

The **BOUNDARY** parameter allows a boundary label to be attached to a face. The properties of such a label are set or modified using the **BOUNDARY** command.

The **ELEMTYPE** parameter allows the user to force the use of quadratic elements when creating the database. Any elements that touch the face will become quadratic if the database is created with mixed surface elements. See “[The SOLVERS Command](#)” on page 3-105 for greater detail.

The **SIZE**, **NORMALTOL** and **SURFACETOL** parameters control the mesh size of elements on and near the face, when generating the surface and volume mesh.

The **LEVEL** parameter controls the storage of data when there is conflict during the merging of multiple faces. The data set with the greater level will be maintained. The result of merging 2 faces with the same level is indeterminate.

## The **FILL** Command

---

**Summary** Fills the model with a mesh of tetrahedral elements.

**Menu Route:** Model↓  
Generate volume mesh

**Command Line Parameters:**

Command	<b>FILL</b>	
Parameter	Default	Function
<b>TOLERANCE</b>	1.0e-6	Tolerance used for checking for equivalent points

**Notes** This command generates the volume mesh and must be called prior to creating the database.

The **TOLERANCE** parameter should not need to be adjusted in most cases, and can control the tolerance used for matching equivalent points.

The cells inside the model body are meshed sequentially. On execution of the **FILL** command, the outline frame of the volume which is being processed is highlighted. A progress bar is displayed at the bottom of the Modeller window.

This command may take a long time for complex models or large meshes.

## The **FILTER** Command

---

**Summary** Sets a filter for graphical selection.

**Icons**



**Menu Route:** Picking

**Command Line Parameters:**

Command	FILTER		
Parameter	Default	Function	
<b>TYPE</b>	<b>NONE</b>	<b>NONE</b>	No graphical selection
		<b>BODY</b>	Sets the type of topological entity that is selected when double clicking over it in the display.
		<b>CELL</b>	
		<b>FACE</b>	
		<b>EDGE</b>	
		<b>VERTEX</b>	
		<b>LCS</b>	
<b>COMMAND</b>	<b>PICK</b>	<b>PICK</b>	Determines the effect of selecting by cursor
		<b>HIDE</b>	
		<b>LIST</b>	
<b>DISPLAY</b>	<b>VISIBLE</b>	<b>VISIBLE</b>	Any visible object can be selected
		<b>EXPLICIT</b>	Only objects explicitly selected for display can be selected

**Notes** The filter command controls the graphical interaction with the Modeller. Setting the **TYPE** parameter controls the type of entity that can be selected using the cursor.

The **COMMAND** parameter controls the effect of double-clicking when an object is highlighted. The **DISPLAY** parameter can be used for extra control of selection of items.

**COMMAND=PICK** allows objects to be added to the list of picked entities by issuing a **PICK** command.

**COMMAND=HIDE** hides the object from the display by issuing a **SELECT OPTION=HIDE** command.

**COMMAND=LIST** generates a list showing details of the object and attached data.

This command does not generate an entry in the history stream, and hence produces no **UNDO** or **REDO** command state.

## The FITTEDCPE Command

---

### Summary

Create or modify fitted constant perimeter end (CPE) conductors.

### icon



### Menu Route:

Create↓

Conductor → Fitted constant perimeter end

Operations↓

Modify conductors → Fitted constant perimeter end

### Command Line Parameters:

Command	FITTEDCPE		
Parameter	Default	Function	
OPTION	NEW	NEW	Creates a new fitted CPE conductor
		MODIFY	Modifies properties of the picked fitted CPE conductors
		LOAD	Loads defaults from picked conductors
DRIVELABEL		Name for the drive label	
LCNAME		Name for Local Coordinate System for coordinate system 1	
SYMMETRY		Rotational symmetry about global Z axis	
X0		Origin of coordinate system 2	
Y0			
Z0			
THETA		Euler angles defining orientation of coordinate system 2	
PHI			
PSI			
RXY		Reflection symmetries in XY, YZ and ZX planes	
RYZ			
RZX			
ALPHA		Azimuthal angular position of the straight	

Command	<b>FITTEDCPE</b> ( <i>continued</i> )		
Parameter	Default	Function	
<b>BETA</b>		Cutter angle	
<b>A</b>		Cross-sectional width	
<b>B</b>		Cross-sectional height	
<b>H1</b>		Half length of the straight	
<b>R1</b>		Radius of forming cylinder	
<b>R2</b>		Radius of cross-over arc	
<b>CURD</b>		Current density in the conductor	
<b>TOLERANCE</b>		Field calculation tolerance	
<b>KEEP</b>	<b>NO</b>	<b>NO</b>	Clear the list of picked items
		<b>YES</b>	Keep the list of picked items for further modification

### Notes

This command creates a new fitted CPE conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked fitted CPE conductors, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked fitted CPE conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked fitted CPE conductors. If the picked fitted CPE conductors do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked fitted CPE conductors to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see “The **CONDUCTOR Sub-command MODIFY**” on page 4-44.

## The **HELICALEND** Command

---

### Summary

Create or modify helical end conductors.

### Icon



### Menu Route:

Create↓  
     Conductor → Helical end  
 Operations↓  
     Modify conductors → Helical end

### Command Line Parameters:

Command	HELICALEND		
Parameter	Default	Function	
<b>OPTION</b>	<b>NEW</b>	<b>NEW</b>	Creates a new helical end conductor
		<b>MODIFY</b>	Modifies properties of the picked helical end conductors
		<b>LOAD</b>	Loads defaults from picked conductors
<b>DRIVELABEL</b>		Name for the drive label	
<b>LCNAME</b>		Name for Local Coordinate System for coordinate system 1	
<b>SYMMETRY</b>		Rotational symmetry about global Z axis	
<b>X0</b>		Origin of coordinate system 2	
<b>Y0</b>			
<b>Z0</b>			
<b>THETA</b>		Euler angles defining orientation of coordinate system 2	
<b>PHI</b>			
<b>PSI</b>			
<b>RXY</b>		Reflection symmetries in XY, YZ and ZX planes	
<b>RYZ</b>			
<b>RZX</b>			
<b>ALPHA</b>		Azimuthal angular position of the straight	

Command	<b>HELICALEND</b> (continued)		
Parameter	Default	Function	
<b>BETA</b>		Cutter angle	
<b>A</b>		Cross-sectional width	
<b>B</b>		Cross-sectional height	
<b>H1</b>		Half length of the straight	
<b>H2</b>		Length of conductor	
<b>R1</b>		Radius of forming cylinder	
<b>R2</b>		Radius of cross-over arc	
<b>CURD</b>		Current density in the conductor	
<b>TOLERANCE</b>		Field calculation tolerance	
<b>KEEP</b>	<b>NO</b>	<b>NO</b>	Clear the list of picked items
		<b>YES</b>	Keep the list of picked items for further modification

### Notes

This command creates a new helical end conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked helical end conductors, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked helical end conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked helical end conductors. If the picked helical end conductors do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked helical end conductors to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see “[Helical Ends](#)” on page 4-32.

## The **HIDE** Command

---

*Summary* Hides picked entities.

*Command Line Parameters:*

Command	<b>HIDE</b>
No parameters	

*Notes* The **HIDE** command hides all picked objects from display, and unpicks them. Objects can be temporarily hidden (without removing them from the list of picked objects) by using **SELECT OPTION=PICKEDREMOVE**.

## The **HISTORY** Command

---

**Summary** Controls the size of the history stream.

**Menu Route:** Edit  
History

**Command Line Parameters:**

Command	<b>HISTORY</b>	
Parameter	Default	Function
<b>STATES</b>	0	Number of history states to keep in the history stream

**Notes**

The **HISTORY** command allows the size of the history stream to be limited. This may help reduce the memory usage of the modeller. The **STATES** parameter determines how many states are kept. If zero, there is no limit. If greater than zero, only this number of operations will be retained, and hence the **UNDO** command can only be called a limited number of times.

## The **IMPORT** Command

---

**Summary** Imports conductor data from a conductor file generated by the Modeller, pre or post processor.

**Menu Route:** Create↓  
Conductor → Import

**Command Line Parameters:**

Command	<b>IMPORT</b>	
Parameter	Default	Function
<b>FILE</b>		Conductor file to be imported

**Notes** The **IMPORT** command reads a file containing conductor data. The conductor file must be generated by the **WRITE** subcommand of the pre processor **CONDUCTOR** command (page 4-48), the **ACTION=EXPORT** option in the **CONDUCTOR** command of the post processor (page 5-39), or by the **EXPORT** command in the Modeller (page 3-55).

The data in the file may contain more than one conductor.

The conductor data in the file contains the Local Coordinate System values, not the names of Local Coordinate Systems. If an existing Local Coordinate System is found that matches the coordinate system values of a conductor, this will be used, otherwise a new Local Coordinate System will be created and set as coordinate system 1 for the conductor.

## The LABEL Command

---

**Summary** Controls setting additional labels on entities.

**Menu Route:** Properties↓  
Additional labels

**Command Line Parameters:**

Command	LABEL		
Parameter	Default	Function	
OPTION	ADD	ADD	Adds a label to picked items
		REMOVE	Removes the named label from the picked items
		CLEAR	Removes all labels from the picked items
LABEL		Label to be added or removed	

**Notes** The LABEL command can be used to add labels to any entity within the model. This label can then be used as the basis for grouping items for selection for visual display, or for picking for modification.

Using **OPTION=ADD** will add the named label to all picked items. **OPTION=REMOVE** will remove the named label from all items that are picked, if they have the label.

**OPTION=CLEAR** will remove all labels from the picked items.

## The **LCS** Command

---

### Summary

Creates a new named Local Coordinate System (LCS), or changes the location of an existing Local Coordinate System of this name.

### Menu Route:

Create↓  
Local coordinate system

### Command Line Parameters:

Command	<b>LCS</b>	
Parameter	Default	Function
<b>LCNAME</b>		Name of Local Coordinate System to be created or modified
<b>X0</b>		Origin of the Local Coordinate System
<b>Y0</b>		
<b>Z0</b>		
<b>THETA</b>		Euler angles defining the local orientation of the Local Coordinate System
<b>PHI</b>		
<b>PSI</b>		

### Notes

All values must be given and are specified in the Working Coordinate System. If a Local Coordinate System of the given name already exists, its position and orientation are reset to the new values.

Each Local Coordinate System is uniquely identified by its **LCNAME**. A Local Coordinate System can be set to be the current working coordinate system using the **WCS** command. A Local Coordinate System is also used within conductors to define the Local Coordinate System 1 of the conductor.

A Local Coordinate System can be picked from the display by setting the **FILTER** command to **TYPE=LCS**. If a single Local Coordinate System has been picked it can be set to be the Working Coordinate System and can also be renamed. If one or more Local Coordinate System is picked they can be transformed, copied or deleted. Copying a Local Coordinate System will generate a new name from the original to keep the name unique.

## The **LIST** Command

---

**Summary** Shows information about entities within the model.

**Menu Route:** Properties↓  
List properties

**Command Line Parameters:**

Command	<b>LIST</b>		
Parameter	Default	Function	
<b>TYPE</b>	<i>none</i>	<b>BODY</b>	Type of entity whose data is to be shown
		<b>CELL</b>	
		<b>FACE</b>	
		<b>EDGE</b>	
		<b>VERTEX</b>	
		<b>LCS</b>	
		<b>CONDUCTOR</b>	
<b>NUMBER</b>		Entity number	

**Notes** The list command shows information about the entity specified by **TYPE** and **NUMBER** parameter. If no entity corresponds to this information, data on all picked entities is given.

This command can be activated graphically by using **FILTER COMMAND=LIST**. Subsequent selections using the cursor will show information on that entity, rather than picking it.

## The **LOAD** Command

---

**Summary** Loads a data from a model file.

**Icon**



**Menu Route:**

```
File↓
  Open model data
Create↓
  Insert from file
```

**Command Line Parameters:**

Command	FACEDATA		
Parameter	Default	Function	
<b>OPTION</b>	<b>NEW</b>	<b>NEW</b>	Clears all existing model data and loads a new model file
		<b>INSERT</b>	Inserts new data from the file into the existing model
<b>FILE</b>		File to be opened	

**Notes**

If opening a new file, **OPTION=NEW**, the existing model data is cleared. At this point the user will be asked to confirm the operation if data has been changed.

**LOAD OPTION=NEW** sets the currently open file. This filename will be changed by the **SAVE OPTION=NEW** command and cleared by the **CLEAR REVERT=NO** commands.

If **OPTION=INSERT**, all bodies, conductors and Local Coordinate Systems within the data file are loaded into the existing model. Only undefined data attached to any of these items is loaded. For example, if the current model has a material label Iron defined, the material properties associated with Iron are not updated from those in the data file.

A filename must be given.

The **LOAD** command will operate on standard *.opc* files and binary *.opcb* files, and will also work on ACIS *.sat* files from the Modeller or other packages. When loading *.sat* files from other software packages, it should be noted that the model may not be well suited to finite element mesh generation and analysis and may cause problems.

The binary form of the data file can be loaded by explicitly giving the *.opcb* extension. The loading of these files may be slightly faster, but the files may not be portable between different platforms. The *.opcb* files should not be renamed to a different file type, since the file type is used to determine the format.

## The **MATERIALS** Command

---

**Summary**                Sets the material properties associated with a material label.

**Menu Route:**            Models↓  
                              Set material properties

**Command Line  
Parameters:**

Command	<b>MATERIALS</b>		
Parameter	Default	Function	
<b>OPTION</b>		<b>PICK</b>	Adds a material label to a list to be set
		<b>UNPICK</b>	Clears the list of material labels to be set
		<b>RESET</b>	Sets picked materials to have the properties of air
		<b>MODIFY</b>	Sets the data for the picked materials
		<b>METRE</b>	Work in SI units
		<b>CGS</b>	Work in CGS units
		<b>INCH</b>	Work in SI units with inches
		<b>MM</b>	Work in SI units with MM
		<b>MICRON</b>	Work in SI units with microns
		<b>LIST</b>	Lists the material properties of the picked materials
	<b>DELETE</b>	Deletes the picked materials	
<b>MATERIALLABEL</b>		Material label used with <b>OPTION=PICK</b>	

Command	<b>MATERIALS</b> ( <i>continued</i> )	
Parameter	Default	Function
<b>ANISOTROPY</b>		Sets the anisotropy of all permeability, permittivity and conductivity
	<b>ISOTROPIC</b>	Set to isotropic
	<b>PACKED</b>	Set to packed
	<b>ANISOTROPIC</b>	Set to anisotropic
<b>LINEARITY</b>		Sets the properties to be linear or non-linear
	<b>LINEAR</b>	Set to linear
	<b>NONLINEAR</b>	Set to non-linear
<b>MUANISOTROPY</b>		Sets the anisotropy of permeability
	<b>ISOTROPIC</b>	Set to isotropic
	<b>PACKED</b>	Set to packed
	<b>ANISOTROPIC</b>	Set to anisotropic
<b>MULINEARITY</b>		
	<b>LINEAR</b>	Set to linear
	<b>NONLINEAR</b>	Set to non-linear
<b>MU</b>		Isotropic linear permeability
<b>HC</b>		Isotropic coercivity
<b>BH</b>		Isotropic non-linear BH curve
<b>MPHASE</b>		Phase lag for isotropic permeability
<b>MUXX</b>		Anisotropic components of linear permeability
<b>MUY Y</b>		
<b>MUZZ</b>		
<b>HCX</b>		Anisotropic components of linear coercivity
<b>HCY</b>		
<b>HCZ</b>		
<b>BHX</b>		Anisotropic components of non-linear permeability
<b>BHY</b>		
<b>BHZ</b>		
<b>MAPHASE</b>		Complex phase lag for anisotropic permeability
<b>SIGANISOTROPY</b>		Sets the anisotropy of conductivity
	<b>ISOTROPIC</b>	Set to isotropic
	<b>ANISOTROPIC</b>	Set to anisotropic
<b>SIGMA</b>		Isotropic conductivity
<b>SPHASE</b>		Phase lag for isotropic conductivity
<b>SIGXX</b>		Anisotropic components of conductivity
<b>SIGYY</b>		
<b>SIGZZ</b>		

Command	<b>MATERIALS</b> (continued)	
Parameter	Default	Function
<b>SAPHASE</b>		Complex phase lag for anisotropic permittivity
<b>EPSANISOTROPY</b>		Sets the anisotropy of permittivity
	<b>ISOTROPIC</b>	Set to isotropic
	<b>ANISOTROPIC</b>	Set to anisotropic
<b>EPSILON</b>		Isotropic permittivity
<b>EPHASE</b>		Phase lag for isotropic permittivity
<b>EPSXX</b>		Anisotropic components of permittivity
<b>EPSYY</b>		
<b>EPSZZ</b>		
<b>EAPHASE</b>		

### Notes

This command defines the material characteristics for use by the analysis programs.

A set of material labels is picked using the command repeatedly, with **OPTION=PICK** and a **MATERIALLABEL** specified. A material label can be removed from the set using **OPTION=UNPICK**. If no **MATERIALLABEL** is given, the set is emptied.

Issuing the command with **OPTION=MODIFY** will modify the properties of the picked material labels to the new values given in the parameters. The value of properties associated with the material labels are unchanged if the parameter is unset.

The default value of each of the parameters is updated to be the common value of all the picked material labels. If the data of one of these parameters is unset, or the picked material labels do not share the same value, then the parameter value is left clear.

New material labels are assumed to be linear and isotropic. **OPTION=RESET** will clear the properties associated with all of the picked material labels.

The working material unit set can be changed using **OPTION=METRE**, **OPTION=CGS** or one of the other sets. If the units are changed, the values in the parameters are converted to the new unit set, and values entered will be interpreted in this unit set.

The properties of all material labels can be listed using **OPTION=LIST**.

Material labels that are not used, i.e. have no cell referencing them can be deleted using **OPTION=DELETE**. Deleting a material label that is in use will reset its properties to those of air.

All parameters can be specified. Which of these values will be used by the analysis will depend upon the analysis module being used, e.g. the phase lag properties are not used by TOSCA.

## The **MESH** Command

---

**Summary** Generates the surface mesh for the model body.

**Menu Route:** Model↓  
Generate surface mesh

**Command Line Parameters:**

Command	<b>MESH</b>	
Parameter	Default	Function
<b>SIZE</b>	1	Maximum element size in the mesh
<b>NORMALTOL</b>	30	Maximum angle between mesh facets normals (degrees)
<b>SURFACETOL</b>	0.0	Absolute tolerance to which the mesh must fit a curved surface
<b>TOLERANCE</b>	1.0e-6	Absolute tolerance used to check point coincidence

**Notes** The mesh command can only be called once the model body has been created using the **MODEL** command.

The operation creates the surface mesh from the information provided. If cells, faces, edges or vertices have their own mesh data attached, the smaller value is used.

The mesh element size on a face will be determined from the distribution of elements along the edges. The mesh element size along an edge is determined from the smallest values from:

- the information from the **MESH** command parameters;
- mesh information on any cells that contain the edge;
- mesh information on any faces that are bounded by the edge;
- mesh information on the edge itself;
- mesh size information on the vertices at the ends of the edge.

The faces of all cells are meshed sequentially. On execution of the **MESH** command, the outline frame of the face being processed is highlighted. A progress bar is also displayed at the bottom of the Modeller.

The **MESH** command may take some time for complex models. On completion the status bar at the bottom of the window displays the message "Surface Mesh".

## The **MODEL** Command

---

### Summary

Creates a single body model suitable for meshing and analysis.

### Menu Route:

Model↓  
     Create model body  
 Model↓  
     Delete model body

### Command Line Parameters:

Command	<b>MODEL</b>		
Parameter	Default	Function	
<b>OPTION</b>	<b>CREATE</b>	<b>CREATE</b>	Creates a model body for meshing and analysis
		<b>DELETE</b>	Deletes the model body

### Notes

The **MODEL OPTION=CREATE** command is used to generate a single body model suitable for meshing. It must be used before a surface or volume mesh can be generated. If bodies are picked, their union is formed without regularisation to form the single body. If no bodies are picked, the union of all bodies in the model is formed without regularisation to form the model body. If any of the bodies have the name **BACKGROUND**, they are intersected with the resulting model body to restrict the model. This is useful when only a section of the model is needed, but when the geometry is easier to build completely, e.g. for axi-symmetric models where the construction is best done using cylinders.

Any cell within the model body, that has material label name **NULL** is deleted from the model body.

Once created, the model geometry cannot be changed. Properties associated with parts of the model can be adjusted, but these changes are not applied to the components used to form the model, and hence changes will be lost when reverting to component view.

When the model body has been created, this is the only body that can be seen. Component bodies used to form the model body still exist, but may not be manip-

ulated or viewed until the model body has been deleted using **MODEL OPTION=DELETE** or with the **DELETE** command.

The model body can be saved, and the component bodies will be saved when using **SAVE OPTION=ALL** or **OPTION=NEW**. When loading, the view will be of the model body, rather than the components. If loading with **OPTION=INSERT**, the model body is not loaded.

Conductors and Local Coordinate Systems are unaffected by the **MODEL** command and can still be created and modified.

The reason for the model command is that when meshing, it is necessary to have a single body, so that all volumes are correctly bounded, and the mesh is continuous throughout the model. However, a single body model makes it difficult to adjust the dimensions and positioning of sections of the model, which is simple when the model is made up of several independent bodies.

The **MODEL** command aims to overcome these two distinct requirements by allowing an easy switch between the two. It creates a special body formed from the non-regular union of the set of picked bodies, or all existing bodies if none are picked. If any body has the name 'background', this body is then intersected with the model body that was formed. This has the effect of limiting the size of the model to that of the background region, and is useful for imposing e.g. quarter symmetry, when the model itself has been constructed in its entirety.

## The **MOUSE** Command

---

*Summary* Swap the functionality of middle and right mouse buttons.

*Icon*



*Command Line Parameters*

Command	<b>MOUSE</b>
No Parameters	

*Notes*

The **MOUSE** command swaps the functionality of the middle and right mouse buttons so that the software can be used with a 2-button mouse.

The default mouse functions in the 3d Graphics Window are:

- left: rotate
- middle: zoom
- right: translate

## The **PERIODICITY** Command

### Summary

Defines the set of transformations for creating periodic symmetry boundary pairings within the model.

### Menu Route:

Model↓  
Symmetry conditions...

### Command Line

#### Parameters:

Command	PERIODICITY		
Parameter	Default	Function	
<b>NSETS</b>	0	Defines the number of periodicity transforms defined.	
<b>DX1</b>	0	Defines the x, y, z displacement of the first periodicity transform.	
<b>DY1</b>	0		
<b>DZ1</b>	0		
<b>ROTX1</b>	0	Defines an axis, based on the Global Coordinate System, about which a rotational component of the first periodicity transform can be defined	
<b>ROTY1</b>	0		
<b>ROTZ1</b>	0		
<b>ANGLE1</b>	0	The angle of rotation about the axis.	
<b>TYPE1</b>	<b>POSITIVE</b>	<b>POSITIVE</b>	Set the potentials on paired faces to be the same.
		<b>NEGATIVE</b>	Set the potentials on paired faces to have the opposite sign.
<b>OPTION1</b>	<b>MATCH</b>	<b>MATCH</b>	Force the geometry of the model to match the specified periodicity transform.
		<b>EXTERNAL</b>	Consider the faces on the external surface of the model when pairing faces.
		<b>SYMMETRY</b>	Consider faces which have been assigned the symmetry boundary condition.

Command	PERIODICITY (continued)	
Parameter	Default	Function
<b>DX2</b>	0	Defines the x, y, z displacement of the second periodicity transform.
<b>DY2</b>	0	
<b>DZ2</b>	0	
<b>ROTX2</b>	0	Defines an axis, based on the Global Coordinate System, about which a rotational component of the second periodicity transform can be defined
<b>ROTY2</b>	0	
<b>ROTZ2</b>	0	
<b>ANGLE2</b>	0	The angle of rotation about the axis.
<b>TYPE2</b>	<b>POSITIVE</b>	See <b>TYPE1</b>
<b>OPTION2</b>	<b>MATCH</b>	See <b>OPTION1</b>
<b>DX3</b>	0	Defines the x, y, z displacement of the third periodicity transform.
<b>DY3</b>	0	
<b>DZ3</b>	0	
<b>ROTX3</b>	0	Defines an axis, based on the Global Coordinate System, about which a rotational component of the third periodicity transform can be defined
<b>ROTY3</b>	0	
<b>ROTZ3</b>	0	
<b>ANGLE3</b>	0	The angle of rotation about the axis.
<b>TYPE3</b>	<b>POSITIVE</b>	See <b>TYPE1</b>
<b>OPTION3</b>	<b>MATCH</b>	See <b>OPTION1</b>

### Notes

The **PERIODICITY** command is used to define periodic or symmetry boundaries. Periodic boundaries are recognised by the TOSCA and SCALA analysis packages.

Periodic boundaries link the potential values, and hence the fields, over one surface to the potential values over another surface within the model. More precisely, the potential values on one surface are set equal to the values on the other surface (with or without a change of sign).

The **PERIODICITY** command is used to define which faces are linked or paired within the model. This pairing is achieved by specifying a periodicity transform which maps one face, defined as the master face, onto its paired face, the slave face. Edges and vertices are also paired by the periodicity transform. A maximum of 3 separate periodicity transforms can be specified. The actual number of sets used is controlled by the **NSETS** parameter.

Each transform is defined by a displacement and a rotation about the origin. For the first periodicity transform the displacement is defined by the parameters **DX1**,

**DY1** and **DZ1**, which correspond to the x, y and z components of the displacement. The rotational component of the transform is specified by the vector **ROTX1**, **ROTY1**, **ROTZ1** and the parameter **ANGLE1**. These define the axis of rotation and the parameter **ANGLE1** determines the angle of rotation about this axis.

The parameter **TYPE1** specifies whether the potentials over paired faces have the same (**TYPE1=POSITIVE**) or opposite (**TYPE1=NEGATIVE**) sign. The parameter **OPTION1** defines the method by which vertices, edges and faces are paired.

- To crop the geometry of the model to match the periodicity conditions defined, **OPTION1=MATCH** can be specified.
- To pair only the external faces, **OPTION1=EXTERNAL**, can be specified. When an exact match is not possible between faces, faces are subdivided to ensure that the paired faces have the same geometry
- To pair faces with the **SYMMETRY** boundary condition applied **OPTION1=SYMMETRY** can be used. This option can be selected when only certain faces on the external surface of the model have to be paired. Only one of the faces to be paired needs to have the **SYMMETRY** boundary condition applied.

The second and third periodicity transforms can be defined in a similar way.

The pairing of faces is performed when the model body is created. Once the model body has been formed, paired vertices, edges and faces are assigned labels which can be displayed. A label is also given to faces with positive symmetry and to faces with negative symmetry.

When the surface mesh is generated, the mesh on paired faces will be identical. A consequence of this is that any surface mesh refinements specified on one vertex, edge or face of a pair will also be applied to the mesh refinement on the second vertex, edge or face of the pair.

Warnings are given if:

- A defined periodicity transform does not result in the pairing of any faces within the model.
- A face with the **SYMMETRY** boundary condition applied is located within the interior of the model. Periodic boundaries can only be defined on the external surfaces of the model.
- A face with the **SYMMETRY** boundary condition is not paired by the periodicity transforms specified.
- A face is paired to more than one other face within the model. Each face can only be paired to one other face.

## The **PICK** Command

---

**Summary** Picks items for modification.

**Icons**



**Menu Route:** Picking

**Command Line Parameters:**

Command	PICK		
Parameter	Default	Function	
OPTION	ADD	RESET	Clears all picked items
		ADD	Adds items into the list of picked objects
		REMOVE	Removes items from the list
		TOGGLE	Adds if the items are not picked, otherwise removes them
		FILTER	Removes any picked entities that are not of the current filter type
		CHANGE	Replaces picked entities with those of the current filter type
		ALL	Picks all entities of the current filter type
PROPERTY		Property of items including MATERIAL, BOUNDARY, ELEMENT, POTENTIAL etc.	
LABEL		Label associated with the specified property type	
TYPE	none	BODY	Type of entity whose data is to be shown
		CELL	
		FACE	
		EDGE	
		VERTEX	
		LCS	
		CONDUCTOR	

Command	<b>PICK</b> ( <i>continued</i> )	
Parameter	Default	Function
<b>NUMBER</b>		Identifier for the item of the entity type
<b>PTU</b>		Coordinate in or near to an entity to be picked
<b>PTV</b>		
<b>PTW</b>		

### Notes

The **PICK** command allows objects to be picked for modification. Objects picked are stored in an ordered list and can then be used in many operations.

The list can be cleared using **OPTION=RESET**.

**OPTION=ADD** adds new entities to the list.

**OPTION=REMOVE** removes entities from the list

**OPTION=TOGGLE** adds an entity if it is not in the list, or removes it if it is in the list.

With **OPTION=ADD**, **OPTION=REMOVE** and **OPTION=TOGGLE** there are 3 ways of specifying items:

- The first uses an entity **TYPE** with a **NUMBER**. This method is used during graphical selection of objects. The entity type can be **BODY**, **CELL**, **FACE**, **EDGE**, **VERTEX**, **LCS** or **CONDUCTOR**. Each object has a unique integer identifier and this is specified in the **NUMBER** parameter. If number is zero, all objects of the given type are picked.

This method should not be used in scripts if possible. After commands such as **COMBINE** or **DELETE**, the numbering of the items will be changed. The method in which the numbering is applied cannot be guaranteed to be the same on subsequent runs, so use of this method may mean that the wrong items are picked if rerunning a script.

- If no number is specified an entity of the given type can be picked by providing a coordinate (**PTU**, **PTV**, **PTW**), specified in the Working Coordinate System. This method will pick a body or cell that contains the point. If the type is face, edge or vertex, the nearest entity will be found. Local Coordinate Systems and conductors cannot be picked using this method. If the pick is ambiguous, no item is picked.
- The third method uses the data attached to each of the entities. The **PROPERTY** parameter can be used to select a type of data item that is stored. For example **PROPERTY=MATERIAL** will pick objects that have a material label attached. This will be all cells, as they all have this information.

By specifying a label the **PICK** command can be more refined, e.g. **PROPERTY=MATERIAL LABEL=AIR**, would pick only cells with a material label of air.

If more than one item is picked in a single **PICK** command, the order in which items are added to the list is undefined. In most cases this is unimportant. For **COMBINE OPERATION=SUBTRACT** or **OPERATION=TRIM**, the order is critical to the result and more care must be taken in picking the objects.

Subsequent **PICK** commands always add picked items to the end of the list of picked items.

Picked faces can be displayed with direction arrows showing the normal direction of the face. This normal direction is used by various commands, e.g. **SWEEP OPTION=DISTANCE**. The direction of the normal can be reversed by double clicking over one of the direction vectors (with **FILTER TYPE=FACE**). This has the effect of generating a **PICK** command with a negative entity **NUMBER**.

**OPTION=FILTER** filters the list of picked items with the current filter entity type, and removes any that do not match.

**OPTION=CHANGE** changes the list of picked items to those of the current filter entity type. If this type is **CELL**, this option will pick all cells in all picked bodies. If type is **FACE**, this option will pick all faces in all picked bodies and cells. The filter option is then applied to remove any entities not of this type.

The **TYPE** parameter of the **PICK** command will be used. If it is unset, the **TYPE** parameter of the **FILTER** command will be used instead.

- **TYPE=CELL** converts all picked bodies to picked cells, and unpicks any other entity type (such as faces, edges and vertices).
- **TYPE=FACE** converts all picked bodies and cells to picked faces, and unpicks any other entity type.
- **TYPE=EDGE** converts all picked bodies, cells and faces to picked edges, and unpicks any other entity type.
- **TYPE=VERTEX** converts all picked bodies, cells, faces and edges to picked vertices, and unpicks any other entity type.

As explained above, the process only works in one direction (from edges to vertices) not the other way.

**OPTION=ALL** selects all items of the filter entity type.

## The **PICTURE** Command

---

**Summary** Saves the current display in a file or to the clipboard.

### Icons



### Menu Route:

```

Edit↓
  Copy to file
Edit↓
  Copy to clipboard
  
```

### Command Line Parameters:

Command	<b>PICTURE</b>		
Parameter	Default	Function	
<b>SAVE</b>	<b>NO</b>	<b>NO</b>	Copy file to clipboard
		<b>YES</b>	Save picture in a file
<b>FILENAME</b>		The name of the file to be saved	
<b>TYPE</b>	<b>PNG</b>	<b>PNG</b>	Type of image file format used to save the picture
		<b>BMP</b>	
		<b>XPM</b>	

### Notes

This command allows the current display to be stored to the clipboard, or in a file.

If **SAVE=NO** the image is placed on the clipboard and can then be pasted into another application. Under UNIX, the image is placed on the X-Selection.

With **SAVE=YES**, the image is stored in a file. The filename can be specified in the **FILENAME** parameters, and the format can be selected with the **TYPE** parameter.

An image can be printed directly using the **PRINT** command (see “The **PRINT** Command” on page 3-90).

## The PRECISIONDATA Command

---

**Summary** Sets tolerances for the Modeller.

**Menu Route:** Edit↓  
Tolerances

**Command Line Parameters:**

Command	PRECISIONDATA	
Parameter	Default	Function
ABSTOL	1.0e-6	The minimum distance between geometric objects within the model
SMALLFACE	1.0e-6	The area of a face that is considered to be of negligible area

### Notes

This command controls the tolerance of operations within the Modeller. These tolerances should be set according to the size of the model being created. In general, these parameters should be set before building the model. Changing the minimum dimension when parts of the model have already been defined (especially if these are now outside the defined range) may cause some operations to fail.

The **ABSTOL** parameter sets the minimum distance at which points are treated as being independent. The **ABSTOL** parameter also defines the maximum dimension of the model. This is set to allow a dynamic range of  $10^{10}$  for double precision data (with allowances for calculation tolerances), giving a maximum dimension of  $10^{10} * \text{ABSTOL}$ , i.e.  $10^4$  by default. If the model does not sit comfortably within this dimension range, this tolerance should be adjusted accordingly.

During some Boolean operations, the tolerancing may be such that very small faces are created. The **SMALLFACE** parameter defines the area considered by the Modeller to be negligible. Any face of area less than this value is unwanted, and the Modeller tries to remove it. In some cases, this may fail and the user should take steps to improve the model. Small faces are likely to cause problems to the mesh generation, and if successful, the mesh is likely to be of lower quality near such a face.

## The **PRINT** Command

---

**Summary** Prints the current display.

**Icon**



**Menu Route:** File↓  
Print

**Command Line  
Parameters:**

Command	<b>PRINT</b>
No parameters	

**Notes**

This command allows the current display to be printed. A dialog of the available printers and their options for printing is opened. The current display can then be printed on the selected printer.

The printers available are determined from the system.

An image can be saved to file or copied to the clipboard using the **PICTURE** command (see “[The PICTURE Command](#)” on page 3-88).

## The **PRISM** Command

---

### Summary

Creates an  $n$ -sided prism or pyramid, with points evenly distributed around a circular or elliptic base.

### Icon



### Menu Route:

Create↓  
Object → Prism/pyramid

### Command Line Parameters:

Command	<b>PRISM</b>	
Parameter	Default	Function
<b>NAME</b>		Attaches this name to the body formed
<b>SIDES</b>		Number of sides on the base of the prism
<b>HEIGHT</b>		Height along the axis of the prism
<b>MAJORRADIUS</b>		Major radius of the base of the prism
<b>MINORRADIUS</b>		Minor radius of the base of the prism
<b>TOPRADIUS</b>		Radius at the top of the prism

### Notes

All values must be given. At least 3 sides must be specified. The **HEIGHT** must be greater than zero.

If the **TOPRADIUS** is given as zero, the body formed is a pyramid.

The points needed to form the  $n$ -sided polygon are evenly positioned around a base ellipse given by the **MAJORRADIUS** and **MINORRADIUS**, and the sides of the polygon are connected by straight edges. The prism is formed by sweeping along the z-axis a distance of **HEIGHT**. If **TOPRADIUS** is different from the major radius of the base, the prism is tapered. If the **TOPRADIUS** is zero, an  $n$ -sided pyramid is formed.

The centre of the body formed is at the origin (i.e. the base is not centred on the origin).

## The **RACETRACK** Command

---

**Summary** Create or modify racetracks.

**Icon**



**Menu Route:** Create↓  
                   Conductor → Racetrack  
 Operations↓  
                   Modify conductors → Racetrack

**Command Line Parameters:**

Command	<b>RACETRACK</b>		
Parameter	Default	Function	
<b>OPTION</b>	<b>NEW</b>	<b>NEW</b>	Creates a new racetrack conductor
		<b>MODIFY</b>	Modifies properties of the picked racetrack conductors
		<b>LOAD</b>	Loads defaults from picked conductors
<b>DRIVELABEL</b>		Name for the racetrack drive label	
<b>LCNAME</b>		Name for Local Coordinate System for coordinate system 1	
<b>SYMMETRY</b>		Rotational symmetry about global Z axis	
<b>X0</b>		Origin of coordinate system 2	
<b>Y0</b>			
<b>Z0</b>			
<b>THETA</b>		Euler angles defining orientation of coordinate system 2	
<b>PHI</b>			
<b>PSI</b>			
<b>RXY</b>		Reflection symmetries in XY, YZ and ZX planes	
<b>RYZ</b>			
<b>RZX</b>			

Command	RACETRACK (continued)		
Parameter	Default	Function	
XP1		Inside lower on the racetrack cross-section	
YP1			
A		Cross-sectional width	
B		Cross-sectional height	
H1		Half-length of the straight	
R1		Radius of the arc	
CURD		Current density in the conductor	
TOLERANCE		Field calculation tolerance	
KEEP	NO	NO	Clear the list of picked items
		YES	Keep the list of picked items for further modification

### Notes

This command creates a new racetrack conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked racetracks, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked racetrack conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked racetracks. If the picked racetracks do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked racetracks to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see “[Racetracks](#)” on page 4-29.

## The **REDO** Command

---

**Summary** Redo previously undone commands

**Icon**



**Menu Route:**

Edit↓  
    Redo  
Edit↓  
    History

**Command Line Parameters:**

Command	<b>REDO</b>	
Parameter	Default	Function
<b>STATE</b>	<i>none</i>	Name of the state to which the history is to be undone

**Notes**

This command allows an **UNDO** command to be reversed, by forwarding the state of the model to a former position.

If no value for **STATE** is given, the operation forwards one state.

The **REDO** facility is available to go to future positions in the history stream after the **UNDO** command. The **REDO** command is available until a new entry is added into the history stream, i.e. a new command changes the current state of the model.

Note: If generating command scripts from log files generated during sessions, it is advisable to remove any **UNDO** or **REDO** commands that specify a state name, as the state name is dependent upon the number of commands issued. This makes it *very sensitive* to changes earlier in the script.

## The **RENAME** Command

---

**Summary** Changes the name on picked bodies or Local Coordinate System.

**Menu Route:** Properties↓  
Rename

**Command Line Parameters:**

Command	<b>RENAME</b>	
Parameter	Default	Function
<b>NAME</b>		Name to be used to replace the existing name

**Notes** The **RENAME** command changes the name associated with the picked bodies or with a single Local Coordinate System.

If more than one Local Coordinate System is selected, the operation has no effect. Two Local Coordinate Systems may not have the same name. Renaming a Local Coordinate System will not affect any conductors that use this Local Coordinate System for describing coordinate system 1.

Bodies with the name **BACKGROUND** define the extent of the model space when creating the model body with the **MODEL** command.

## The **SAVE** Command

---

**Summary** Saves model data to file.

**Icon**



**Menu Route:**

```
File↓
  Save
  Save as new model data
  Save model with mesh
Operations↓
  Save picked components...
  Export picked bodies
```

**Command Line**

**Parameters:**

Command	SAVE		
Parameter	Default	Function	
<b>OPTION</b>	<b>ALL</b>	<b>ALL</b>	Saves all model data to the current file
		<b>NEW</b>	Creates a new file containing all the model data
		<b>PICKED</b>	Saves all picked bodies, Local Coordinate Systems and conductors
		<b>EXPORT</b>	Exports picked bodies to <i>sat</i> format file
		<b>MESH</b>	Saves all model data and mesh data in binary format
<b>FILE</b>		File name to be opened	
<b>VERSION</b>	6.0	SAT file version for exporting	

**Note**

**SAVE** will save a file containing all the model data. A saved file can be opened with the **LOAD** command (see page 3-71).

To save all the model data in a new *.opc* file, use **SAVE OPTION=NEW**. To overwrite a previously saved or loaded *.opc* file, use **SAVE OPTION=ALL**. Note that

**OPTION=ALL** can only be used once the open file has been established by **LOAD OPTION=NEW** or **SAVE OPTION=NEW** commands.

To save a subset of the model, pick the bodies, Local Coordinate System or conductors and use **OPTION=PICKED** to save them to file.

An ACIS *.sat* file of any valid version number can be exported using **SAVE OPTION=EXPORT**.

To save the whole model and the finite element mesh in a binary (*.opcb*) file, use **SAVE OPTION=MESH**. This does not change the open file name used by **SAVE OPTION=ALL**.



Command	SELECT (continued)		
Parameter	Default	Function	
TYPE	none	BODY	Type of entity whose data is to be shown
		CELL	
		FACE	
		EDGE	
		VERTEX	
		LCS	
		CONDUCTOR	
NUMBER		Identifier for the item of the entity type	
PTU		Point nearest to or contained within an entity of the selection TYPE	
PTV			
PTW			
WAIT	NO	YES	Do not update the display from the selection change
		NO	Update the display from this selection
AUTOUPDATE	YES	YES	Refresh the display after every change
		NO	Never update the display
SELECTLEVEL	1	Set the options from which display items can be chosen	

### Notes

The **SELECT** command allows a list of objects to be selected for display and a list of objects to be explicitly hidden. It provides facilities for viewing the different data attached to entities, differentiating between different data with different colours. It also allows parts of the model to be hidden. This is a stronger option than not being displayed, as it forces items to be hidden from the display.

**OPTION=DEFAULT** will clear all display items and reset the data to display any entities with properties of *Material*, *Coiltype*, *LCName* or *System*.

The selected and hidden items are completely cleared using **OPTION=RESET**.

**OPTION=ADD** selects an item or data characteristic for display.

**OPTION=HIDE** selects an item or data characteristic to be explicitly hidden from the display.

**OPTION=REMOVE** removes the display or hide selection from an item or data characteristic.

With **OPTION=ADD**, **OPTION=REMOVE** or **OPTION=HIDE**, there are three ways of specifying the selection:

- Data is attached to each of the entities. The **TYPE** parameter can be used to select a type of data item that is stored. For example **PROPERTY=BOUNDARY** will select objects that have a boundary label attached.

By specifying a label the **SELECT** command can be refined, e.g. adding **LABEL=FACE1** in the above example would restrict the selection to entities with attached boundary condition label **FACE1**.

When using **OPTION=REMOVE** with only the **PROPERTY** parameter set, the selection of all individually selected items in that property list will also be removed.

It should be noted that different entity types will be selected by different properties. For example the *Material* property is attached to cells, so selecting **PROPERTY=MATERIAL** will select cells. The *boundary* property is attached to faces, so **PROPERTY=BOUNDARY** will select faces.

The colour with which each object is displayed will indicate the specific label of the property. Where the colour for the display of a section is ambiguous, the part is displayed with the specific colour indicating the ambiguity. Colours can be changed using the **COLOUR** command (see [page 3-39](#)).

- The second method for selection uses an entity **TYPE** with a number. This method is used with graphical selection when hiding an entity. The entity type can be **BODY**, **CELL**, **FACE**, **EDGE**, **VERTEX**, **LCS** or **CONDUCTOR**. Each object has a unique integer identifier and this number is specified in the **NUMBER** parameter. If **NUMBER** is not given, all objects of the given type are selected.

It is not recommended that this method be used in scripts. After operations such as **COMBINE** or **DELETE**, the numbering of the items will be changed. The method in which the numbering is applied cannot be guaranteed to be the same on subsequent runs or versions of the software, so use of this method may mean that the wrong items are selected.

- A third method uses an entity **TYPE** and a point coordinate (**PTU**, **PTV**, **PTW**), defined in the current Working Coordinate System. Only entities of type body, cell, face, edge or vertex can be selected by this method. For cells and bodies, the entity that contains the point is selected. For others, the entity nearest to the supplied point is selected. If there is more than one possible selection, the result is undefined.

**OPTION=UNHIDE** unhides all objects that have been hidden by the second and third methods described above. Objects hidden by **PROPERTY** or **LABEL** will not be affected.

The different selections are stored and can be used in any combination. For example,

```
SELECT OPTION=ADD TYPE=MATERIAL
```

will display all entities with material data (i.e. all cells). A second command of

```
SELECT OPTION=HIDE TYPE=MATERIAL LABEL=AIR
```

will mean that all materials are displayed except those with material label **AIR**. A further command of

```
SELECT OPTION=HIDE TYPE=POTENTIAL LABEL=REDUCED
```

will mean that any reduced potential regions are also excluded from the display.

The faces of objects are used to display the objects, i.e. bodies, cells and faces are displayed by drawing their faces. This means that the selection of the display is quite complex, as the choice of selected and hidden items will often conflict. The following rules are used to choose the colour and faces to be shown:

1. Highlight any picked item.
2. Determine any other parts of the model that have been hidden and flag these for exclusion from the display.
3. Include faces with associated data that have been selected for display, which are not hidden and which are not already displayed.
4. Include bodies with associated data that have been selected for display, which are not hidden and which are not already displayed.
5. Include cells with associated data that have been selected for display, which are not hidden and which are not already displayed.

It can be seen from the above order that the display will always include all picked items. **OPTION=PICKEDREMOVE** can be used to override this, so that any hidden object is temporarily removed from the display. Subsequent displays after issuing this command will revert to the normal display mode with picked items included.

A further option, **SELECTLEVEL=1** restricts the options to the basic information attached to the different entities. More options, e.g. the boundary condition type that is associated to a boundary label, are available with **SELECTLEVEL=2**.

By default any change to the model will cause the display to be refreshed. This behaviour can be changed using **AUTOUPDATE=NO**. In such a case, the **THREED** command must be called to refresh the display to show changes. This should not normally be used during an interactive session as it will mean there is no visual feedback of any actions performed.

When making multiple selections the **WAIT** parameter allows the **AUTOUPDATE** to be overridden until the next change. The **WAIT** parameter is always reset to **NO**, so must be explicitly declared with any **SELECT** command when needed.

## The **SOLENOID** Command

---

**Summary** Create or modify solenoids.

**Icon**



**Menu Route:**

Create↓  
     Conductor → Solenoid  
 Operations↓  
     Modify conductors → Solenoid

**Command Line Parameters:**

Command	SOLENOID		
Parameter	Default	Function	
OPTION	NEW	NEW	Creates a new solenoid conductor
		MODIFY	Modifies properties of the picked solenoid conductors
		LOAD	Loads defaults from picked conductors
DRIVELABEL		Name for the solenoid drive label	
LCNAME		Name for Local Coordinate System for coordinate system 1	
SYMMETRY		Rotational symmetry about global Z axis	
X0		Origin of coordinate system 2	
Y0			
Z0			
THETA		Euler angles defining orientation of coordinate system 2	
PHI			
PSI			
RXY		Reflection symmetries in XY, YZ and ZX planes	
RYZ			
RZX			

Command	SOLENOID <i>(continued)</i>		
Parameter	Default	Function	
XP1		First point on the solenoid cross-section	
YP1			
XP2		Second point on the solenoid cross-section	
YP2			
XP3		Third point on the solenoid cross-section	
YP3			
XP4		Fourth point on the solenoid cross-section	
YP4			
CP1		Curvatures of the 4 sides of the cross-section	
CP2			
CP3			
CP4			
CURD		Current density in the conductor	
TOLERANCE		Field calculation tolerance	
KEEP	NO	NO	Clear the list of picked items
		YES	Keep the list of picked items for further modification

### Notes

This command creates a new solenoid when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked solenoids, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked solenoid conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked solenoids. If the picked solenoids do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked solenoids to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see [“Solenoids” on page 4-27](#).

## The **SOLVERS** Command

---

### Summary

Creates a new database, or adds a new simulation to an existing database.

### Menu Route:

Model↓  
Create analysis database

### Command Line Parameters:

Command	<b>SOLVERS</b>		
Parameter	Default	Function	
<b>PROGRAM</b>	<b>TOSCAMAGN</b>	Analysis program type	
		<b>TOSCAMAGN</b>	TOSCA: magnetic, electrostatic or current flow
		<b>TOSCAELEC</b>	
		<b>TOSCACURRENT</b>	
		<b>SCALA</b>	SCALA: Space charge
		<b>ELEKTRASS</b>	ELEKTRA: Steady state harmonic, transient or velocity (linear and rotational)
		<b>ELEKTRATR</b>	
		<b>ELEKTRAVL</b>	
		<b>ELEKTRARO</b>	
		<b>SOPRANOSS</b>	SOPRANO: Steady state harmonic or eigenvalue
<b>SOPRANOEV</b>			
<b>FILE</b>		Name of the database file	
<b>OPTION</b>	<b>NEW</b>	<b>NEW</b>	Create a new database file
		<b>ADD</b>	Add a new simulation to an existing database file
<b>UNITS</b>	<b>CGS</b>	Specify the unit set to be used to write the database	
		<b>CGS</b>	CGS units
		<b>METRE</b>	SI units
		<b>MM</b>	SI units with MM
		<b>MICRON</b>	SI units with microns
		<b>INCH</b>	SI units with inches

Command	<b>SOLVERS</b> <i>(continued)</i>		
Parameter	Default	Function	
<b>ELEMENT</b>	<b>MIXED</b>	Type of elements to be created	
		<b>MIXED</b>	Use element type defined by cell property
		<b>LINEAR</b>	Use all linear elements
		<b>QUADRATIC</b>	Use all quadratic elements
<b>SURFACE</b>	<b>CURVED</b>	Type of elements to be created in all elements that touch surfaces	
		<b>MIXED</b>	Use element type defined by face property
		<b>CURVED</b>	As mixed, but curved surfaces default to quadratic
		<b>LINEAR</b>	Use all linear elements
		<b>QUADRATIC</b>	Use all quadratic elements

### Notes

The solvers command creates a new database or adds a new simulation to an existing database. When adding a simulation to a database, the **UNITS**, **ELEMENT** and **SURFACE** parameters are not used.

The type of analysis to be used is set by the **PROGRAM** parameter.

The type of elements are controlled by the **ELEMENT** and **SURFACE** parameters. The **ELEMENT** parameter allows the type of volume elements in cells to be controlled. The **SURFACE** parameter allows the type of volume elements that touch a face to be controlled.

## The **SPHERE** Command

---

**Summary**            Creates a sphere.

**Icon**



**Menu Route:**        Create↓  
                          Object → Sphere

**Command Line  
Parameters:**

Command	<b>SPHERE</b>	
Parameter	Default	Function
<b>NAME</b>		Attaches this name to the body formed
<b>X0</b>		X coordinate of centre
<b>Y0</b>		Y coordinate of centre
<b>Z0</b>		Z coordinate of centre
<b>RADIUS</b>		Radius of the sphere

**Notes**                All values must be given. A sphere of the given **RADIUS** is formed, centred on the coordinate (**Z0**, **Y0**, **Z0**).

The coordinates specified are in the Working Coordinate System.

## The **STRAIGHT** Command

---

**Summary** Create or modify straights.

**Icon**



**Menu Route:** Create↓  
                   Conductor → Straight bar  
 Operations↓  
                   Modify conductors → Straight bar

**Command Line Parameters:**

Command	<b>STRAIGHT</b>		
Parameter	Default	Function	
<b>OPTION</b>	<b>NEW</b>	<b>NEW</b>	Creates a new straight conductor
		<b>MODIFY</b>	Modifies properties of the picked straight conductors
		<b>LOAD</b>	Loads defaults from picked conductors
<b>DRIVELABEL</b>		Name for the drive label	
<b>LCNAME</b>		Name for Local Coordinate System for coordinate system 1	
<b>SYMMETRY</b>		Rotational symmetry about global Z axis	
<b>X0</b>		Origin of coordinate system 2	
<b>Y0</b>			
<b>Z0</b>			
<b>THETA</b>		Euler angles defining orientation of coordinate system 2	
<b>PHI</b>			
<b>PSI</b>			
<b>RXY</b>		Reflection symmetries in XY, YZ and ZX planes	
<b>RYZ</b>			
<b>RZX</b>			
<b>A</b>		Cross-sectional width	

Command	STRAIGHT (continued)		
Parameter	Default	Function	
<b>B</b>		Cross-sectional height	
<b>H1</b>		Length of the straight	
<b>CURD</b>		Current density in the conductor	
<b>TOLERANCE</b>		Field calculation tolerance	
<b>KEEP</b>	<b>NO</b>	<b>NO</b>	Clear the list of picked items
		<b>YES</b>	Keep the list of picked items for further modification

### Notes

This command creates a new straight conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked straights, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked straight bar conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked straights. If the picked straights do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked straights to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see “[Straight Bars](#)” on page 4-35.

## The **SWEEP** Command

---

### Summary

Sweeps a picked face through space to form a new body, or to extend the existing body owning the face.

### Menu Route:

Operations↓  
Sweep face

### Command Line Parameters:

Command	SWEEP		
Parameter	Default	Function	
TYPE	DISTANCE	DISTANCE	Sweep a fixed distance along the face normal
		VECTOR	Sweep along a specified vector
		ROTATION	Sweep a fixed angle about an axis
DISTANCE		Distance to be swept through, <b>TYPE=DISTANCE</b> only	
DU		Vector to sweep along, <b>TYPE=VECTOR</b> only	
DV			
DW			
ROTU		Axis of rotation, <b>TYPE=ROTATION</b> only	
ROTV			
ROTW			
POSU		Point on the axis of rotation, <b>TYPE=ROTATION</b> only	
POSV			
POSW			
ANGLE		Angle of rotation, <b>TYPE=ROTATION</b> only	
REGULARISE	NO	NO	Leaves the starting face
		YES	Removes the original face if part of a body

<b>DRAFTTYPE</b>	<b>NONE</b>	<b>NONE</b>	No draft during the sweep
		<b>ANGLE</b>	The area of the swept face increases with the draft angle
<b>DRAFTANGLE</b>			The draft angle

**Notes**

Only a single planar face may be swept in the command.

The sweep is controlled by the **TYPE** parameter.

If **TYPE=DISTANCE**, the face is swept along its normal. Vectors representing the normal direction can be viewed on picked faces with the **VECTOR** command. This normal direction can be toggled by picking one of these vectors.

**TYPE=VECTOR** will sweep along the specified vector.

**TYPE=ROTATION** allows sweeping about an angle around an axis passing through a point.

For **TYPE=VECTOR** and **TYPE=ROTATION**, care should be taken to avoid sweeping a face where the normal to the face is almost perpendicular to the direction.

Care should also be taken to avoid sweeping a face into another part of the same body. This can cause errors in the body formed. Some of these difficulties may be resolved, but other operations may progress and form inconsistent bodies. Such bodies will be highlighted using the **CHECK** command, but cannot be easily corrected.

When sweeping, the face can be changed in area by using the draft angle.

The coordinates specified are in the Working Coordinate System.

## The **TANGENTIALCPE** Command

### Summary

Create or modify tangential constant perimeter end (CPE) conductors.

### Icon



### Menu Route:

Create↓

Conductor → Tangential constant perimeter end...

Operations↓

Modify conductors → Tangential constant perimeter end...

### Command Line Parameters:

Command	<b>TANGENTIALCPE</b>		
Parameter	Default	Function	
<b>OPTION</b>	<i>none</i>	<b>NEW</b>	Creates a new tangential CPE conductor
		<b>MODIFY</b>	Modifies properties of the picked tangential CPE conductors
		<b>LOAD</b>	Loads defaults from picked conductors
<b>DRIVELABEL</b>		Name for the drive label	
<b>LCNAME</b>		Name for Local Coordinate System for coordinate system 1	
<b>SYMMETRY</b>		Rotational symmetry about global Z axis	
<b>X0</b>		Origin of coordinate system 2	
<b>Y0</b>			
<b>Z0</b>			
<b>THETA</b>		Euler angles defining orientation of coordinate system 2	
<b>PHI</b>			
<b>PSI</b>			
<b>RXY</b>		Reflection symmetries in XY, YZ and ZX planes	
<b>RYZ</b>			
<b>RZX</b>			
<b>ALPHA</b>		Azimuthal angular position of the straight	

Command	TANGENTIALCPE (continued)		
Parameter	Default	Function	
BETA		Cutter angle	
A		Cross-sectional width	
B		Cross-sectional height	
H1		Half length of the straight	
R1		Radius of forming cylinder	
R2		Radius of cross-over arc	
CURD		Current density in the conductor	
TOLERANCE		Field calculation tolerance	
KEEP	NO	NO	Clear the list of picked items
		YES	Keep the list of picked items for further modification

### Notes

This command creates a new tangential CPE conductor when using **OPTION=NEW**.

**OPTION=MODIFY** it is used to operate on the list of picked tangential CPE conductors, changing geometry, coordinate systems, symmetries, current density, tolerance etc. The current common values of the picked tangential CPE conductors are available by calling **CONDUCTOR OPTION=LOAD**.

The default values of the command parameters are updated to match common values shared by all of the picked tangential CPE conductors. If the picked tangential CPE conductors do not have a common value for a parameter, that parameter is left unset. Using **OPTION=MODIFY** will change the conductor data of all of the picked tangential CPE conductors to the new values given in the parameters, but will not affect the conductor data of any unset parameter.

For more information on the parameters, see [“Constant Perimeter Ends” on page 4-34](#).

## The **THREED** Command

### Summary

Control the display of the geometry.

### Icons



### Menu Routes

View↓  
 Refresh  
 Set view...  
 Views

### Command Line Parameters

Command	<b>THREED</b>		
Parameter	Default	Function	
<b>OPTION</b>		Command option:	
		<b>GETVIEW</b>	Retrieve view parameters after mouse interaction.
		<b>INIT</b>	Refresh picture without changing the view.
<b>SETVIEW</b>		Refresh picture using the view parameters.	
<b>SIZE</b>		Display extends from the origin by <b>SIZE</b> in each direction. <b>SIZE=0</b> requests the initial view of the model	
<b>ROTX</b>	20	Rotation of model around X axis.	
<b>ROTY</b>	20	Rotation of model around Y axis.	
<b>ROTZ</b>	0	Rotation of model around Z axis.	
<b>XORIGIN</b>	0	X coordinate at centre of picture	
<b>YORIGIN</b>	0	Y coordinate at centre of picture	
<b>ZORIGIN</b>	0	Z coordinate at centre of picture	
<b>PERSPECTIVE</b>	<b>YES</b>	Perspective switch:	
		<b>YES</b>	Perspective view.
		<b>NO</b>	Orthographic view.

### Notes

The **THREED** command updates the 3d picture of the model. The picture consists of the three dimensional geometry of the model and conductors. The view can be adjusted using the mouse buttons or by setting explicitly using this command.

- **OPTION=GETVIEW**: updates the values of the parameters **SIZE**, **ROTX**, **ROTY**, **ROTZ**, **XORIGIN**, **YORIGIN** and **ZORIGIN**.
- **OPTION=SETVIEW**: uses the current values of **SIZE**, **ROTX**, **ROTY**, **ROTZ**, **XORIGIN**, **YORIGIN** and **ZORIGIN**.
- **OPTION=REFRESH**: updates the picture without changing the view. If automatic update is switched off using the command **SELECT AUTOUPDATE=NO**, this command option must be called to update the display.

Perspective view can be switched off using **PERSPECTIVE=NO**.

## The **TITLE** Command

---

**Summary** Add title, date and time to the display.

**Menu Route** View↓  
Title

### Command Line Parameters

Command	<b>TITLE</b>		
Parameter	Default	Function	
<b>STRING</b>	<i>none</i>	A graphics window title.	
<b>POSITION</b>	<b>TOPLEFT</b>	Title position:	
		<b>BOTTOMCENTRE</b>	Bottom centre
		<b>BOTTOMLEFT</b>	Bottom left
		<b>BOTTOMRIGHT</b>	Bottom right
		<b>NONE</b>	No title
		<b>TOPCENTRE</b>	Top centre
		<b>TOPLEFT</b>	Top left
		<b>TOPRIGHT</b>	Top right
<b>DATE</b>	<b>TOPLEFT</b>	Time/date position:	
		<b>BOTTOMCENTRE</b>	Bottom centre
		<b>BOTTOMLEFT</b>	Bottom left
		<b>BOTTOMRIGHT</b>	Bottom right
		<b>NONE</b>	No date and time
		<b>TOPCENTRE</b>	Top centre
		<b>TOPLEFT</b>	Top left
		<b>TOPRIGHT</b>	Top right

**Notes** The **TITLE** command controls the display of a title and the date and time. There is a choice of 6 positions for each. If the same position is chosen for both the title and the date, the title appears above the date and time.

## The **TORUS** Command

---

**Summary**            Creates a torus.

**Icon**



**Menu Route:**        Create↓  
                              Object → Torus

**Command Line  
Parameters:**

Command	<b>TORUS</b>	
Parameter	Default	Function
<b>NAME</b>		Attaches this name to the body formed
<b>X0</b>		X coordinate of centre
<b>Y0</b>		Y coordinate of centre
<b>Z0</b>		Z coordinate of centre
<b>MAJORRADIUS</b>		Major radius of the torus
<b>MINORRADIUS</b>		Minor radius of the torus

**Notes**                All values must be given. A torus with the given radii is formed, centred on the coordinate **(Z0,Y0,Z0)**.

The coordinates specified are in the Working Coordinate System.

## The **TRANSFORM** Command

---

**Summary** Transforms or copies bodies, Local Coordinate Systems and conductors.

**Menu Route:**  
 Operations↓  
     Transform  
 Operations↓  
     Copy

**Command Line Parameters:**

Command	TRANSFORM		
Parameter	Default	Function	
OPTION	APPLY	APPLY	Apply the transformation to the picked items
		COPY	Creates a copy of the new item and applies the transformation to it
TYPE	DISPLACE	DISPLACE	Use a displacement transformation
		ROTATE	Use rotation about an axis
		REFLECT	Reflect in a plane
		SCALE	Apply anisotropic scaling
		EULER	Rotate by Euler angle set
DU	0	Translation component in the working coordinate system	
DV	0		
DW	0		
ROTU	0	Rotation axis in the working coordinate system	
ROTV	0		
ROTW	1		
ANGLE	0	Angle of rotation about the given axis	
NU	0	Normal vector to the plane of reflection	
NV	0		
NW	1		
SCU	1	Scaling factors in the working coordinate system	
SCV	1		
SCW	1		

Command	TRANSFORM (continued)		
Parameter	Default	Function	
THETA	0	Values of Euler angle rotations	
PHI	0		
PSI	0		
LABEL		Adds a label to each item transformed or copied.	
COUNT	1	Number of copies to be created	
KEEP	NO	NO	Reset the list of picked objects
		YES	Keep the list of picked objects

### Notes

The **TRANSFORM** command is used to transform the geometry of existing items, or to create a copy and transform this.

The **TYPE** of transformation determines which of the parameter values are used. All others are ignored. The **COUNT** parameter is only used when copying, and allows multiple copies to be produced from a single command.

Type	Parameters
DISPLACE	DU, DV, DW
ROTATE	ROTU, ROTV, ROTW, ANGLE
REFLECT	NU, NV, NW
SCALE	SCU, SCV, SCW
EULER	THETA, PHI, PSI

If bodies are picked, they can be transformed using **OPTION=APPLY**, or **OPTION=COPY** can be used to generate a copy of these bodies that are then transformed.

If any Local Coordinate Systems or conductors are picked, the **SCALE** and **REFLECT** transformations will not have any effect on these objects, as they must remain with the same handedness and size.

Picked cells, faces, edges and vertices are unaffected by transformations.

If only cells have been picked, the **OPTION=COPY** operation can be used to create a new body for each cell that is copied.

If only faces have been picked, the **OPTION=COPY** operation can be used to create a new body for each face. This body then contains only a copy of this face. The body formed will enclose no volume, but the face may be selected and used in sweep or other operations.

When transforming or copying conductors, the operator modifies either Local Coordinate System 1 or Local Coordinate System 2 of the conductor:

- If Local Coordinate System 1 is the current Working Coordinate System, Local Coordinate System 2 is changed.
- If Local Coordinate System 1 is not the Working Coordinate System, a copy of Local Coordinate System 1 is created and transformed, and the conductor's Local Coordinate System 1 is changed to point to the new Local Coordinate System.

The transformed items will all have the **LABEL** attached as a user label. If copying, the original items will be given the label. All new items created by copying will be given the label suffixed by a count from 1 to **COUNT**, i.e. labeltext1, labeltext2 etc. This allows the items created by copying to be easily grouped according to the number of times that the transformation was applied to the original items.

The **KEEP** parameter can be used if more than one transformation is to be applied to the picked set of objects, to avoid needing to re-pick them.

## The UNDO Command

---

**Summary** Rewinds the history to a specified position.

**Icon**



**Menu Route:**

```
Edit↓
  Undo
Edit↓
  History
```

**Command Line Parameters:**

Command	<b>UNDO</b>	
Parameter	Default	Function
<b>STATE</b>	<i>none</i>	Name of the state to which the history is to be undone

**Notes**

This command returns the state of the model to a former position.

If no value for **STATE** is given, the operation rewinds one previous state.

Most operations add an entry to the history stream. This entry is given a unique **STATE** name (based upon the command name and the number of commands so far). Any state name within the list can be specified in the **STATE** parameter. This allows the model to be returned to any previous state by a single command.

After an **UNDO** command, a **REDO** facility is available to go back to future positions in the history stream (see “[The REDO Command](#)” on page 3-94). The **REDO** command is available until a new entry is added into the history stream.

Note: If generating command scripts from log files generated during sessions, it is advisable to remove any **UNDO** or **REDO** commands that specify a state name, as the state name is dependent upon the number of commands issued. This makes it *very sensitive* to modifications earlier in the script.

## The **VARIABLE** Command

---

**Summary** Sets a user defined variable.

**Menu Route:** Create ↓  
Variables

**Command Line Parameters:**

Command	<b>VARIABLE</b>		
Parameter	Default	Function	
<b>OPTION</b>	<b>CONSTANT</b>	<b>CONSTANT</b>	Value is constant
		<b>PARAMETER</b>	Expression is stored for later evaluation
		<b>DELETE</b>	Deletes the named variables
		<b>LIST</b>	Lists all stored variables
<b>NAME</b>	<i>none</i>	Variable name	
<b>VALUE</b>	<i>none</i>	Value or expression assigned to the variable	

### Notes

This command creates and changes the values of user defined variables. Such variables must begin with the '#' character and be a total length of not more than 16 characters.

The command replaces the use of the **\$CONSTANT** and **\$PARAMETER** commands, and allows history facilities for **UNDO/REDO**, as well as storage of the variables when loading and saving models.

**OPTION=DELETE** removes a stored variable, although this may not be possible if the variable forms part of an expression which is in use in the Modeller.

**OPTION=LIST** shows the current set of variables and their values.

The **\$CONSTANT** and **\$PARAMETER** commands can be used, but the effects are not stored immediately within the history, so cannot be explicitly undone.

These variables can be used as part of any expression, and so can help to automate command scripts or to assist in model definition by allowing common values to be defined.

## The VECTOR Command

---

**Summary** Displays vectors of attached data and properties on the model.

**Menu Route:** View↓  
Vectors

**Command Line Parameters:**

Command	VECTOR		
Parameter	Default	Function	
COMPONENT	NONE	A	Magnetic vector potential of a face boundary condition
		CONDUCTOR	Conductor current
		FACEDIRECTION	Direction of faces for use with sweeping and emitters
		INA	Incident magnetic vector potential of a face boundary condition
		J	Current density set in the volume property of a cell
		NONE	No vectors will be displayed
		ORIENTATION	Local orientation set in the volume property of a cell
		V	Linear velocity set in the volume property of a cell
SCALE	1	Scale factor for sizing the vectors	
MAGNITUDE	YES	NO	Display all vectors with <b>LENGTH=SCALE</b>
		YES	Display vectors with their magnitude set by <b>COMPONENT*SCALE</b>

**Notes**

This command displays vectors of the **COMPONENT** on displayed objects in the model. If an object has data containing the component, then the vectors of the data will be displayed on it. Other objects will be displayed normally.

The vectors displayed can be scaled in size by the value of **SCALE**. Those vectors that have a magnitude can be displayed either with size given by the magnitude of the **COMPONENT** multiplied by **SCALE** (**MAGNITUDE=YES**), or with a constant size of **SCALE** (**MAGNITUDE=NO**).

## The **VERTEXDATA** Command

---

**Summary** Sets properties of picked vertices.

**Menu Route:** Properties↓  
Vertex properties

### Command Line Parameters:

Command	VERTEXDATA		
Parameter	Default	Function	
<b>OPTION</b>	<b>MODIFY</b>	<b>MODIFY</b>	Applies new values to the picked vertices
		<b>RESET</b>	Clears all data from the picked vertices
<b>SIZE</b>	See notes	Mesh control size	
<b>LEVEL</b>	See notes	Data storage level for the vertex data	

### Notes

This command is used to set or clear the properties of all picked vertices. Vertices initially have no data assigned to them.

If issuing the command with **OPTION=MODIFY**, the new value of parameters that have been set replace the existing values of data on the vertices. The value of data, associated with any of the unset parameters, is left unchanged.

The default value of each of the parameters is updated to be the common value of all the picked vertices. If the data of one of these parameters is unset, or the picked vertices do not share the same value, then the parameter value is left clear.

**SIZE** sets the maximum element side length of any element touching that vertex.

Upon issuing the command with **OPTION=MODIFY**, all picked items are deselected. Issuing the command with **OPTION=RESET** keeps the same set of picked objects.

The **LEVEL** parameter controls the storage of data when there is a conflict during the merging of multiple cells. The data set with the greater level will be maintained. The result of merging 2 cells with the same level is indeterminate.

## The **VOLUME** Command

---

**Summary** Sets the properties associated with a volume label.

**Menu Route:** Model↓  
Set volume properties

**Command Line Parameters:**

Command	<b>VOLUME</b>	
Parameter	Default	Function
<b>OPTION</b>		<b>PICK</b> Adds a volume label to a list of labels to be set
		<b>UNPICK</b> Clears the list of volume labels to be set
		<b>RESET</b> Clears the data from the picked labels
		<b>MODIFY</b> Sets the data for the picked labels
		<b>LIST</b> Lists the properties of picked labels
		<b>DELETE</b> Deletes the picked labels
<b>VOLUMELABEL</b>		Volume label to be picked for modification
<b>VX</b>		Components of linear velocity for ELEKTRA-VL
<b>VY</b>		
<b>VZ</b>		
<b>JX</b>		Components of volume source current density.
<b>JY</b>		
<b>JZ</b>		
<b>THETA</b>		Components of orientation for anisotropic material properties
<b>PHI</b>		
<b>PSI</b>		
<b>PACKING</b>		Packing factor for packed materials
<b>ROTATION</b>		Rotational velocity
<b>CHARGE</b>		Volume charge density

**Notes**

This command defines the volume characteristics for use by the analysis programs.

A set of volume labels is picked using the command repeatedly, with **OPTION=PICK** and a **VOLUMELABEL** specified. A volume label can be removed from the set using **OPTION=UNPICK**. If no **VOLUMELABEL** is given, the set is emptied.

Issuing the command with **OPTION=MODIFY** will modify the properties of the set of picked volume labels to the new values given in the parameters. The value of a property associated with the volume labels is unchanged if the parameter is unset.

The default value of each of the parameters is updated to be the common value of all the picked volume labels. If the data of one of these parameters is unset, or the picked volume labels do not share the same value, then the parameter value is left clear.

**OPTION=RESET** will clear the properties associated with all of the picked volume labels.

The properties of all volumes with the label **VOLUMELABEL** can be listed using **OPTION=LIST**.

Volume labels that are not used, i.e. have no cell referencing them can be deleted using **OPTION=DELETE**. Deleting a volume label that is in use will reset its properties.

All parameters can be specified. Which of these values will be used by the analysis will depend upon the analysis module being used, e.g. the velocity components will only be used by ELEKTRA-VL.

## The **WCS** Command

---

**Summary** Sets the Working Coordinate System (WCS).

**Menu Route:** Operations↓  
Set working coordinate system

**Command Line Parameters:**

Command	<b>WCS</b>		
Parameter	Default	Function	
<b>OPTION</b>		<b>SET</b>	Set Working Coordinate System to Local Coordinate System given by <b>LCNAME</b>
		<b>PICKED</b>	Set Working Coordinate System to be the picked Local Coordinate System
		<b>UNSET</b>	Use the Global Coordinate System as the Working Coordinate System
<b>LCNAME</b>		Name of the Local Coordinate System to be set as Working Coordinate System	

### Notes

This command controls the Working Coordinate System. The Working Coordinate System is a coordinate system in which many commands operate, and it can be used to assist in the definition of geometries that do not lie neatly in the Global Coordinate System. The commands affected are those that create new objects, i.e. **BLOCK**, **CYLINDER**, **PRISM**, **SPHERE**, **TORUS** and **LCS**. It also operates with the **TRANSFORM** and **SWEEP** commands when specifying positions, vectors and axes of rotation.

Local Coordinate Systems can be defined with the **LCS** command. If **OPTION=SET**, a value must be given for **LCNAME** and it must correspond to the name of a Local Coordinate System within the model.

If a single Local Coordinate System has been picked, the **OPTION=PICKED** will set it to be the current Working Coordinate System.

**OPTION=UNSET** will unset the Working Coordinate System, so that the Global Coordinate System will be used.

## The **WINDOW** Command

---

**Summary** Show or hide parts of the display.

### Icons



**Menu Route** View  
Parts of the display

### Command Line Parameters

Command	<b>WINDOW</b>	
Parameter	Default	Function
<b>AXES</b>	<b>YES</b>	Show coordinate axes: <b>YES</b> or <b>NO</b>
<b>SOLID</b>	<b>YES</b>	Show solid view of model: <b>YES</b> or <b>NO</b>
<b>OUTLINE</b>	<b>YES</b>	Show outline view of model: <b>YES</b> or <b>NO</b>
<b>VECTORS</b>	<b>YES</b>	Show vectors on the surface of the model: <b>YES</b> or <b>NO</b>
<b>MESH</b>	<b>YES</b>	Show surface mesh <b>YES</b> or <b>NO</b>

The **WINDOW** command can be used to hide or show again parts of the display which exist. For example, if a surface mesh exists, it can be hidden using **WINDOW MESH=NO** or shown again using **WINDOW MESH=YES**.



# Chapter 4

## OPERA-3d Pre Processor

### Introduction

---

---

The OPERA-3d pre processor can prepare data for the electromagnetic field analysis programs including CARMEN, ELEKTRA, SCALA, SOPRANO and TOSCA. The analysis programs use finite elements to model three dimensional electromagnetic devices, with the added facility for direct evaluation of fields from conductors carrying prescribed currents. The pre processor provides full support for all the features of the analysis programs.

The OPERA-3d pre processor is an interactive program that is used to create and edit three dimensional finite element models, define material characteristics for non-linear magnetic or dielectric components, assign boundary conditions, specify complicated conductor geometries with prescribed excitation, show the models using wire frame and hidden surface displays and output data files in the formats accepted by the analysis programs.

When the pre processor starts, on UNIX operating systems, the first input the user must give is 'device nomination'. This tells the program what graphics options should be used. This is described with the **DEVICE** command (page 4-95) which can be used at any time to reset the device, or change graphics options.

After device nomination, or when the OPERA-3d pre processor is restarted with the **CLEAR** command, the program looks for a file called *oppre.comi* in the current file directory. If such a file exists it is read into the program as a **\$ COMINPUT** file (page 2-22). This allows the user to reset the default values of certain commands, e.g. **COLOUR** (page 4-22) or define frequently used **\$ CONSTANTS** and **\$ PARAMETERS** (page 2-19) each time the program is started.

Two methods of command and data entry are available:

1. Menu system or GUI (Graphical User Interface) - command selection and data specification are carried out under mouse control.
2. Command line input - command selection and data specification are carried out from the keyboard.

Under normal operation, the pre processor starts in the GUI mode.

## The Graphical User Interface

---

The GUI is built from 8 types of input window which are selected and controlled by pointing with the cursor and clicking a mouse button. Some input windows accept characters typed at the keyboard. The input windows are:

- **Horizontal menu:** Only used for top level menu.
- **Vertical menu:** For selecting commands and options.
- **ParameterBox:** For entering numerical or character data.
- **DialogBox:** Combination of text inputs and switches.
- **FileBox:** For selection of files.
- **CDBox:** For selection of current directory or folder.
- **ColourBox:** For re-defining colours.

The GUI also uses MessageBoxes to display messages and questions on the display.

### Menus

Menus are horizontal or vertical lists of keywords which indicate actions to be performed. Menu items are selected by pointing with the mouse and clicking its left button. When the mouse is pointing at a menu item, that item is highlighted.

Alternatively, menu items can be chosen using the keyboard arrow keys: ← and → for a horizontal menu or ↑ and ↓ for a vertical menu. When the required item is highlighted, it can be selected using the <Enter>, <Return> or ↵ keys.

Selecting a menu item can have one of several effects; the action is indicated by a symbol at the right-hand side of the item:

Symbol	Action
↓	Drop Down: this activates a sub-menu. It only exists in the top-level horizontal menu.
	Pull Right: activates a sub-menu.
	Pick and Pull: activates a sub-menu after a selection from the displayed model (see Pick below).
	Return: returns to higher-level menu.

Symbol	Action ( <i>continued</i> )
	Toggle: swaps between 2 options and the symbol changes between  and  . The current state of the program is displayed.
	Option: chooses one from a set of options. The current choice is indicated by  (filled with red).
	Pick: must be followed by a selection from the displayed model. This is done by positioning the cursor (which changes shape to $\oplus$ ) over the required part of the model and pressing the left mouse button.
	Rubber-box: must be followed by selection of diagonally opposite corners of a rectangle. This is done by pressing the left mouse button with the cursor at one corner and dragging the mouse, with the button held down, releasing it at the opposite corner. The menus are automatically hidden while the rubber-box is being used.
	Action: executes a command or requests additional information via a ParameterBox or both; sometimes the menu will close after the specified action.

The **<Esc>** key can be used to escape from a menu without any actions. If the menu does not allow Pick operations, selecting from a higher level menu can also be used to close it.

Not all menu items can be used at all times. For example, it is not possible to execute post processing commands if there is no solution or no mesh. Unavailable menu items are displayed in pale-blue rather than white until they become available as the result of other commands.

## Parameter Boxes

ParameterBoxes (Figure 4.1) are used to input information from the keyboard.



Figure 4.1 A Typical ParameterBox

Default values (if they exist) are displayed and are initially highlighted. When a value is highlighted it can be replaced by the first characters typed. A value can be edited by moving the text cursor before typing any characters keys.

Most ParameterBoxes have **Accept** and **Dismiss** below the list of parameters. These can be selected with the mouse to execute or escape from the command. The mouse can also be used to identify a parameter to be edited.

Editing parameter values and controlling the execution of the command can be achieved with the following keys:

- $\uparrow$ ,  $\downarrow$  and **<Tab>** can be used to move between the parameters.
- $\leftarrow$  and  $\rightarrow$  move the text cursor within the value being edited.
- **<Enter>**, **<Return>** or  $\downarrow$  move to next parameter. If **Accept** is highlighted or there is only one parameter, the command is executed.
- **<Esc>** escapes from the command.
- **<Back-space>** or **<Delete>** delete characters.

To toggle insert mode: **<Insert>** (Windows) or function keys **<F2>** or **<PF2>** (X-windows)

- To move to start of the value **<Home>** (Windows) or function keys **<F3>** or **<PF3>** (X-windows)
- To move to end of the value: **<End>** (Windows) or function keys **<F4>** or **<PF4>** (X-windows)

## FileBoxes and CDBoxes

FileBoxes (Figure 4.2) are used for selecting a file name for reading or writing.

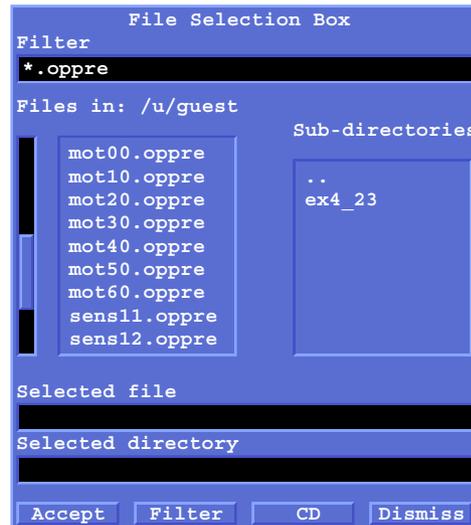


Figure 4.2 A File Selection Box

The FileBox contains a filter string which specifies a subset of all files. If the filter is edited, it can be acted on by typing **<Enter>**, **<Return>** or **↵** or by selecting the **Filter** button.

If there are more files than can be displayed in the FileBox a scroll-bar is displayed at the left side of the list of names. Similarly, if the longest file name is too long for the FileBox a scroll bar is displayed below the list of names. The size of the slider within the scroll bar indicates the proportion of the text which is displayed. The text can be scrolled in two ways:

- clicking above or below the slider in the vertical scroll bar, or to the left or right of the slider in the horizontal scroll bar, scrolls the text by one page in the direction indicated.
- dragging the slider, by pressing and holding the left mouse button while moving the mouse scrolls the text in the direction indicated while the mouse is moving.

One file should be selected from the list of files. Double-clicking (selecting the file twice in quick succession) confirms the selection. Alternatively, the selection can be confirmed by selecting the **Accept** button. The required file name can also be typed into the selection box and accepted by typing **<Enter>**, **<Return>** or **↵**.

The current directory or folder name is shown and its sub-directories are also displayed in a second selection area. The current directory can be changed by using a double-click selection in the same way as for a file name, or by typing the directory name into the selection box and typing **<Enter>**, **<Return>** or **↵**. Any change of directory in a FileBox is remembered for the next time the FileBox is used, unless the current directory is changed using the CDBox which resets the directories for all FileBoxes.

The CDBox implements the Change Directory command within the GUI. It displays a list of subdirectories, which can be selected by double-click or typing in the same way as within the FileBox. If the new directory includes a disk or device name, it can only be selected by typing the full name into the selection box. When the current directory is as required, the CDBox can be closed by typing **<Esc>** or by selecting the  button.

In FileBoxes and CDBoxes, file tree-names and directory names can be given using environment variables (UNIX and Windows only). Environment variables **\$VFDIR** (on UNIX systems) and **%VFDIR%** (on Windows systems) are defined by the software as the parent directory of the software.

## DialogBoxes

DialogBoxes are used to input information using a combination of keyboard and mouse operations. Within a DialogBox there can be:

- Text-inputs: black rectangles. Initially the first text-input is selected and any characters typed will appear there. Any text-input can be selected with the mouse before typing. The **<Enter>**, **<Return>**, **↵**, **↓** or **<Tab>** keys can be used to move to the next text-input. The **↑** key can be used to move to the previous text input. Within a text-input the value can be edited using the editing and function keys defined for ParameterBoxes (see [page 4-4](#)). Selecting a down arrow button  to the right of a text input activates a FileBox to supply a file name for the text-input (see [page 4-5](#)).
- Switches: small squares or 'radio-buttons'. The switches can be on  (red) or off  (blue). The state of a switch can be changed by selecting with the mouse pointer. Turning on one switch might turn others off if the options are mutually exclusive.
- Buttons: labelled rectangles. These are used to action the selected options, or exit without issuing a command. A button can be selected using the mouse or, if it is high-lighted, with the **<Enter>**, **<Return>** or **↵** key. The **<Esc>** can also be used to exit without issuing a command.

- Scrolling lists: a list of items from which one or many can be selected using the mouse. If the list is long or wide, scroll-bars can be used to view other parts of the list in the same way as for FileBoxes (page 4-5).

It is important to remember that DialogBoxes often contain more items than are needed at a particular time. Only those items required should be selected.

## MessageBoxes

MessageBoxes are used by the GUI to communicate important information to the user. There are 5 types of MessageBox each containing black text on a grey background.

- Information: the results of commands, warnings etc. These boxes are labelled with a large **i**. Information MessageBoxes can be dismissed by typing any key on the keyboard (except <F1>) or with the left mouse button.
- If the quantity of information exceeds the size of the window, a scroll-bar is displayed to enable the whole message to be viewed. The window will show the top of the message as the scroll is generated, but other parts of the message can be viewed by dragging the scroll bar up or down.
- Errors: these include Pick operations outside the model space and bad values in ParameterBoxes. The program gives the user another chance to perform the input if an error occurs. Error MessageBoxes are labelled with a large **!**. They can be dismissed by typing any key on the keyboard (except <F1>) or with the left mouse button.
- Questions: these always require a choice between **YES** and **NO**. QuestionBoxes are labelled with a large **?**. They can be dismissed by selecting either the **YES** box or the **NO** box.
- Input: these always require additional information to be given by the user via a ParameterBox, a DialogBox or a FileBox which appears below the MessageBox. The boxes disappear when the information has been supplied.
- Timer: these indicate how much of an operation has been completed. Timers are only displayed for operations for which the estimated elapsed time is greater than 5 seconds. Timer boxes cannot be dismissed, but will automatically disappear when the operation is complete.

## ColourBoxes

The ColourBox (Figure 4.3) is used to redefine colours used on the display. It con-

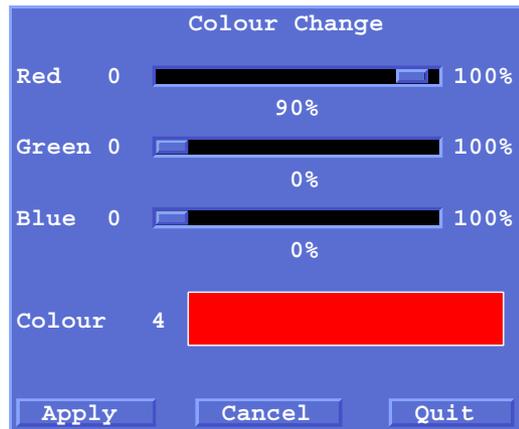


Figure 4.3 A ColourBox

sist of 3 horizontal slider bars, one each for red, green and blue, a square showing the colour as it is changed and three buttons, **Accept**, **Cancel** and **Quit**.

The colour can be adjusted by moving the sliders in two ways:

- clicking to the left or right of the slider decreases or increases the amount of a colour by 10%.
- dragging a slider by pressing and holding the left mouse button and moving the mouse to the left or right decreases or increases the amount of a colour while the mouse is moving in proportion to the distance between the cursor and the slider.

The buttons **Accept** a colour by changing the display to use the colour in the colour square; **Cancel** the changes by restoring the colour to what it was when the ColourBox was opened; and **Quit** the ColourBox.

Note that on some types of display, the new colours are not shown until the picture is redrawn.

## Hiding and Leaving the Menus

Sometimes it is necessary to hide the menus so that the complete picture can be seen. This can be done using the <F1> function key. bring the menus back again.

To leave the menus completely the top-level menu item **MENU - OFF** should be selected. To return to menu-mode, the keyboard command ^ should be given. (This is the caret character followed by <Enter>, <Return> or ↵.)

## Pre Processor Quick Reference Guide

---

The following is a complete list of the ‘top-level’ keyboard commands (which can be entered in response to the prompt ‘OPERA >’) and menu items. Following sections contain complete descriptions of all the keyboard commands, sub-commands and modes of the OPERA-3d pre processor. The commands are described in alphabetical order. The modes of the **DEFINE** command are described in the order they occur when using the program.

### Commands for Keyboard Entry – Command Line

Command line entry is carried out in the text window.

- Help Command:

<b>HELP</b>	Obtain System overview, help on command interpreter, Euler angles, units and new features in this version.
-------------	--

- Mesh definition and editing commands:

<b>DEFINE</b>	Enter the define sequence to input points, facets, subdivisions, extrusions, materials and boundary conditions of the finite element mesh. <b>DEFINE</b> can also be used to define geometry of 20 node ‘brick’ conductor elements.
<b>REDEFINE</b>	Remove all extrusions and enter <b>DEFINE</b> sequence to modify base-plane and redefine extrusions.
<b>EXTEND</b>	Add more extrusion layers to a mesh.
<b>MODIFY</b>	Modify mesh data: points, subdivisions, materials and boundary conditions.
<b>LABEL</b>	Add extra labels to points, lines, facets and volumes.
<b>MATERIALS</b>	Define material properties.
<b>TRANSFORM</b>	Transform a labelled set of points
<b>SLIP</b>	Add or remove a slip surface for CARMEN models.
<b>CHECK</b>	Examine the shape of the finite element volumes, adding label <b>DEBUG</b> . Find the exterior facets of the finite element volumes, adding label <b>EXTERNAL</b> . Report whether hexahedral meshing is possible.

- Conductor definition and editing command:

<b>CONDUCTOR</b>	Enter conductor sub-command mode to define, modify and list conductor data.
------------------	---

- Finite element mesh commands:

<b>MESH</b>	Generate a mesh on the surfaces of the volumes.
<b>FILL</b>	Generate a mesh inside each volume.

- Display command:

<b>DISPLAY</b>	Display some or all of the mesh and conductors in 3D with optional hidden surface removal.
<b>THREED</b>	Start or update the 3d Viewer

- Material B-H characteristic, definition and editing command:

<b>BHDATA</b>	Enter the BH data definition and editing mode.
---------------	--

- Commands to read and write data files:

<b>READ</b>	Read a file of OPERA-3d pre processor data.
<b>EDIT</b>	Read a file of OPERA-3d pre processor data allowing data to be edited at break points.
<b>WRITE</b>	Write a file of OPERA-3d pre processor data for use with <b>READ</b> and <b>EDIT</b> commands.
<b>TABLE</b>	Write a table file containing all the nodes of the model.
<b>SOLVERS</b>	Prepare an analysis database.
<b>IDEAS</b>	Read a finite element mesh from an I-DEAS universal file.

- Program management commands:

<b>DEVICE</b>	Reset or change graphics device.
<b>COLOUR</b>	Enquire and set colours for the display.
<b>TITLE</b>	Control screen titles.
<b>CLEAR</b>	Clear program data and re-initialize all commands.
<b>DUMP</b>	Write a picture file containing the current display.
<b>END</b>	End OPERA-3d pre processor.

## Menu System – GUI

The following is an overview of the first level menu structure. A brief description of the functionality of each item is also given and the page number of the section of this chapter which describes it in detail.

FILE↓	OPTIONS↓	DISPLAY↓	HELP↓	MESH↓	MODIFY↓	DEFINE↓	MENU_OFF
-------	----------	----------	-------	-------	---------	---------	----------

**FILE**↓

File Options		
Read pre processor file	read a pre processor command file	page 4-149
Write pre processor file	write a pre processor command file	page 4-179
I-DEAS universal file	read a finite element mesh from an I-DEAS universal file	page 4-118
Analysis	Create/edit an analysis database	page 4-153
... create new database	Create an analysis database	
... use existing database	Edit an analysis database	
Start analysis now	Start an analysis program using <b>OPERAANL</b>	page 2-28
Write node table file	Write a file in table format containing all the node coordinates	page 4-170
Commands in	read a command input file	page 2-22
Dump picture	send current graphics display to a file	page 4-104
Change directory	change the current directory	page 2-27
System command	run an operating system command	page 2-27
Return	close menu	
End OPERA-3d/Pre	end preprocessing session	page 4-112

**OPTIONS**↓

Options		
Calculator...		
Parameters	set a user defined parameter	page 2-19
Constants	set a user defined constant	page 2-19
List variables	list user defined variables	page 2-19
Colour settings	modify colours in graphics palette	page 4-22
Clear and Reset	clear all mesh data and restart	page 4-21
Dump picture	send current graphics display to a file	page 4-104
Graphics output	select type of graphics output	page 4-95
Title options	specify title options	page 4-176
Title position	specify title position in window	page 4-176
Display title	enter text of title	page 4-176
Return	close menu	

**DISPLAY**↓

3d Viewer...		
... style	specify wire frame, solid etc. display of model	page 4-174
... select parts	specify components to be displayed	page 4-172
... refresh display	start or update 3d Viewer	page 4-171
Display Command ...		
... view	specify display size and orientation of model	page 4-100
... style	specify wire frame, solid etc. display of model	page 4-102
... select parts	specify components to be displayed	page 4-101
... refresh display	clear and re-draw graphics window	page 4-97
... copy 3d view	copy view parameters from 3d Viewer	page 4-100
Return	close menu	

**HELP**↓

Help		
System Overview	general description of pre processor	page 4-117
The GUI	assistance with menu system	
Command line	assistance with command line entry	
Return	close menu	

**MESH**↓

Mesh Generator		
Surface mesh ...		
... options	set options for surface mesh generator	page 4-133
... triangles	generate a surface mesh of triangles	page 4-133
... quadrilaterals	generate a surface mesh of quadrilaterals	page 4-133
Volume mesh ...		
... options	set volume mesh generator options	page 4-115
Mesh	generate a volume mesh	page 4-115
Return	close menu	

**MODIFY**↓

Data Modification		
Mesh number	specify mesh to be modified	page 4-135
No 3d Viewer	choose whether to use 3d Viewer	page 4-135
Point coordinates	modify point coordinates	page 4-135
Subdivisions	modify subdivisions	page 4-135
Material properties	modify element/potential/material types	page 4-135
Boundary conditions	modify boundary conditions	page 4-135
Add slip surface	add slip surface to CARMEN models	page 4-152
Remove slip surface	remove slip surface	page 4-152
Labels	modify entity labels	page 4-125
Transform labelled points	apply transformation to a group of points	page 4-178
Conductors	modify conductor definitions	page 4-24
Return	close menu	

**DEFINE**↓

Data Definition		
Define new mesh	start creating a new mesh	page 4-49
Extend existing mesh	continue extending an existing mesh	page 4-113
Redefine a mesh	remove extrusions, modify baseplane and re-extrude	page 4-151
Check a mesh	check existing mesh geometry	page 4-18
Conductors	create a conductor	page 4-24
Read conductor data	read a file of conductor definition commands	page 4-149
Material properties	define material properties	page 4-129
BH tables	define/edit a B–H curve	page 4-15
Return	close menu	

## The **BHDATA** Command

---

**Menu Route:** DEFINE↓  
BH tables

**Command Line Parameters:**

Command	<b>BHDATA</b>		
Parameter	Default	Function	
<b>UNITS</b>	<b>CGS</b>	The units of B and H	
		<b>CGS</b>	gauss and oersted
		<b>SI</b>	tesla and ampere metre <sup>-1</sup>

The TOSCA, ELEKTRA and CARMEN programs use material characteristics to relate flux density and field intensity of all materials (except for **AIR** and **NULL**). For soft magnetic materials these should be defined in the first quadrant, with the first values of B and H both zero. The curve should not extend beyond saturation magnetisation; the program extrapolates correctly. True hysteresis cannot be modelled, but a limited form, particularly applicable to ferrites, is available in ELEKTRA-SS using complex permeability in the **MATERIALS** command (page 4-129) or the **MATERIAL** sub-command of **SOLVERS** (page 4-164).

Anisotropic materials can be treated as a stack of laminations or by use of multiple BH curves. For hard magnetic materials the demagnetisation curve in the second and third quadrant is used. The orientation of laminations, anisotropic materials and permanent magnets is set by the **VECTOR** property, set in Materials Definition Mode of the **DEFINE** command (page 4-49). The **PACKING** factor of laminations is set there also.

In electrostatics and current flow problems, similar curves can be used to relate the electric field and displacement current or electric field and current density.

The **BHDATA** command is used to create, edit or check tables of pairs of values that define the non-linear BH, DE or JE characteristics of magnetic, dielectric or conducting materials. There must be at least 5 and not more than 50 entries in each table. Material characteristics are not stored with the models in pre processor data files. After a table has been defined or edited the new data should be stored in a materials library file. A separate system directory could be created to contain all such files. Such a directory is supplied with the software and can be accessed

using the directory name  $\$VFDIR/bh$  (UNIX) or  $\%VFDIR%\bh$  (Windows) in the FileBox Selected directory input box.

Names of BH files are requested by the **MATERIALS** sub-command of the **SOLVER** command for all material names used (page 4-164).

The **BHDATA** command has one parameter, **UNITS**, which allows the user to define the data in either **CGS** units (gauss and oersted) or **SI** units (tesla and ampere metre<sup>-1</sup>). Files of BH data stored in one set of units can be loaded and edited in either set of units.

On entering the **BHDATA** command, the user is presented with a default BH curve. In general it will be necessary to delete this curve and replace it with the desired characteristic. In order to use the default curve it must be also stored in a file.

The **BHDATA** command has 8 sub-commands allowing addition of points on the current curve, modification of existing points, access to files, and data checking. The commands do not have named parameters but rely on keywords and values being supplied in the correct order. In the description below the keywords are given in upper case and the variable values in lower case. Keywords can be abbreviated to single letters. In any case only the first 4 characters are decoded.

In the **REPLACE** sub-command, the system variables **BVAL** and **HVAL**, the values of the current point, can be used in expressions. For example, an existing curve can be degraded in a command loop such as:

```
OP-BH > $ do #i 1 10
OP-BH > repl #i h+(b-h)*0.9 h -redr
OP-BH > $ end do
```

The boolean **-REDRaw** prevents the graph of B against H being re-drawn each time round the loop.

## BHDATA Sub-commands

The sub-commands of the **BHDATA** command are described in the following table:

Sub-commands	Function
<b>ADD B H</b>	Add a new point to the end of the table. <b>B</b> and <b>H</b> are numeric values of <b>B</b> and <b>H</b> .
<b>CHECK M N</b>	Check the data and display the interpolations of the data used in analysis. <b>M</b> and <b>N</b> specify the first and last point displayed.
<b>DELETE M N</b>	Deletes the points <b>M</b> to <b>N</b> of the curve. <b>N</b> can have the value <b>*</b> to indicate the last point.
<b>INSERT NUMBER B H</b>	Inserts a new point after the point <b>NUMBER</b> of the curve. <b>B</b> and <b>H</b> are numeric values of <b>B</b> and <b>H</b> .
<b>LOAD FILE</b>	Loads a curve from a file. The file name extension <i>.bh</i> is added to the name if no extension is given. Any points already defined are deleted.
<b>QUIT</b>	Leave the <b>BHDATA</b> command.
<b>REPLACE N B H</b>	Replaces the <b>N</b> th point of the curve. <b>B</b> and <b>H</b> are the new values of <b>B</b> and <b>H</b> . <b>± REDRAW</b> can be used to control whether the curve is re-displayed.
<b>STORE FILE</b>	Stores the curve in a file. The file name extension <i>.bh</i> is added to the name if no extension is given.

## The CHECK Command

---

### Menu Route:

DEFINE↓  
Check a mesh

### Command Line Parameters:

Command	CHECK	
Parameter	Default	Function
MESH	1	Number of the finite element mesh to be checked.
VERBOSE	NO	Verbose reporting of errors:
		NO YES

The **CHECK** command examines the shape of the volumes of the **MESH** number given, adding the label **DEBUG** to any which may cause errors in the mesh generator or analysis programs. It also gives the label **EXTERNAL** to all facets which are not shared by two volumes. There is no checking between facets of different meshes.

The type of checking depends on whether the model is capable of being meshed with hexahedra or can only be meshed with tetrahedra. Hexahedral meshes can be generated in models which obey the following rules:

- All volume facets must be triangles or quadrilaterals.
- The subdivisions on opposite lines of quadrilateral facets must be the same.
- The subdivisions of two lines of triangular facets must be the same.

The **CHECK** command first examines the model to see whether it can be meshed with hexahedra. It then checks each volume as described below.

In verbose mode, the unique labels assigned to volumes have the form **VNNN**, where **NNN** is the internal volume number. This allows individual volumes to be displayed.

If the **CHECK** command is used again, after correcting the volumes in error, the label **DEBUG** is removed from any volumes previously in error and the unique labels, **VNNN** are also removed.

- Example: to display volumes with label **DEBUG** in layer 3:

```
OPERA > check
OPERA > disp type=volu labe=debug ll=3 3
```

## Checks on Hexahedral Models

If the model can be meshed with hexahedra, the following checks are performed.

The program calculates the determinant of the Jacobean which defines the isoparametric mapping from a hexahedron to a unit cube. For the mapping to be unique this must be greater than zero everywhere within the hexahedron. The Jacobean determinant is evaluated at every point defining the volume. In some circumstances the checks may be more strict than necessary. For example, a quadratic volume is checked using quadratic shape functions, but in analysis, the volume may be discretised into linear elements in which the errors do not show up.

If the Jacobean determinant changes sign within a volume, this may be due to any of the errors listed below:

- faces which cut through each other.
- mid-side points which are too close to the corners (closer than 0.25 times length of side).
- excessive curvature or distortion.

The Jacobean can also be used to evaluate the size of each volume. If the Jacobean goes to zero or is negative throughout a volume, this is also reported.

The number of volumes in error is reported for each layer and since they have the label **DEBUG** they can be displayed separately using the **DISPLAY** command.

If **+VERBOSE** is selected, the **CHECK** command reports an error message and assigns a unique label to each **DEBUG** volume. The errors reported are:

- Jacobean changes sign.
- Very small volume.
- Negative volume.
- Mid-side node too close to corner.

The **CHECK** command also counts the total numbers of nodes and elements in the mesh. These numbers are approximate if degenerate volumes (with facets with less than 4 sides) or **NULL** materials are used.

## Checks on other Models

For models with facets with more than 4 sides or irregular subdivision, the following checks are performed.

The equation of the plane of each facet with more than 4 sides is calculated. An error is reported if any points defining the facet do not lie in the plane.

## The **CLEAR** Command

---

### *Menu Route:*

OPTIONS↓  
Clear and Reset

### *Command Line Parameters:*

Command	<b>CLEAR</b>
No Parameters	

The **CLEAR** command puts OPERA-3d back to the state it was in when it first started. It deletes all the data including conductors and construction lines, re-initializes all variables and sets all parameters back to their default values.

User variables are not deleted.

## The COLOUR Command

---

**Menu Route:**      OPTIONS↓  
                          Colour settings

**Command Line Parameters:**

Command	COLOUR	
Parameter	Default	Function
OPTION	ENQUIRE	Option: CONFIGURE, ENQUIRE or SET.
		CONF Re-configure the colour map for a different number of distinct MATERIALS.
		ENQU Enquire which colour numbers are used for each material and part of the display.
		SET Reset colour number CODE to new values of RED, GREEN and BLUE.
CODE	none	Colour map number to be redefined.
RED	none	Amount of red for colour CODE.
GREEN	none	Amount of green for colour CODE.
BLUE	none	Amount of blue for colour CODE.
MATERIALS	5	Number of distinct material colours.

The COLOUR command informs the user which colours are used for each part of the display and enables the definitions of the colours to be changed.

There are three OPTIONS: CONFIGURE, ENQUIRE and SET.

- **CONFIGURE** sets the number of distinct material colours to be made available. The operation of this option depends on the total number of colours available on the display being used. A minimum of 5 and a maximum of 100 distinct material colours are allowed. If there are more materials than colours, the colours are re-used cyclically for higher material numbers. Increasing the number of material colours reduces the number of light-source shading levels.
- **ENQUIRE** lists the colour numbers used for each part of the display.
- **SET** displays the definition of the colour number given by **CODE** in terms of its red, green and blue components. The colour can be changed by giving values to **RED**, **GREEN** and **BLUE** which should each be in the range 0 to 1. The

current values of **RED**, **GREEN** and **BLUE** can be used in expressions to define the new values.

- If the colour **CODE** to be reset is the main colour for air, conductors or a material, etc., the corresponding light-source shading colours are reset to different shades of the new colour.
- On some types of display, the new colours will not be shown until the picture is refreshed.

## The **CONDUCTOR** Command

---

**Menu Route:** DEFINE↓  
Conductors

**Command Line Parameters:**

Command	<b>CONDUCTOR</b>
No Parameters	

This command enters the conductor sub-command mode of OPERA-3d. In this mode new conductors may be defined, existing conductors edited or erased, the parameters of the current set of conductors may be listed or they may be stored in a *cond* data file. The sub-commands are:

<b>CONDUCTOR Sub-commands</b>	
Sub-command	Function
<b>DEFINE</b>	Define a new conductor
<b>ERASE</b>	Erase one or several conductors
<b>MODIFY</b>	Modify one or several conductors
<b>PRINT</b>	Print details of one or several conductors to the terminal and the log file.
<b>QUIT</b>	Leave the <b>CONDUCTOR</b> command.
<b>WRITE</b>	Write a data file with commands to define all the conductors.

The programs have a wide range of pre-defined conductor geometries, ranging from simple solenoids to bedsteads and racetracks wound on the surface of cylinders. There are also primitive conductor elements which can be joined together to build up conductor circuits.

The following conductors are available:

<b>Conductor Shapes</b>	
<b>SOLENOID</b>	Solenoid around $Y'$ axis.
<b>GSOLENOID</b>	Generally orientated solenoid.
<b>RACETRACK</b>	Racetrack around $Y'$ axis.
<b>GRACETRACK</b>	Generally orientated racetrack.
<b>BEDSTEAD</b>	Bedstead around $Y'$ axis.

<b>Conductor Shapes</b> ( <i>continued</i> )	
<b>GBEDSTEAD</b>	Generally orientated bedstead.
<b>HELIX</b>	Helical end racetrack.
<b>CPEND</b>	Constant perimeter end racetrack.
<b>ARC</b>	Circular arc element.
<b>STRAIGHT</b>	Straight bar element.
<b>BR8</b>	8 noded brick element.
<b>BR20</b>	20 noded brick element.

To enable conductors to be oriented in space correctly, local coordinate systems can be defined. To reduce the amount of data necessary, symmetry and reflection codes can be used to replicate a basic shape.

Parameters common to all conductors including the local coordinate systems and replication parameters will be described first, before details of the conductor shapes are given and lastly the parameters of the sub-commands are listed.

## Current Density, Drive Label and Tolerance

The current flowing in the conductors is defined by the current density (**CURD**). For conductors with changing cross section the current density applies to the first face of the conductor. For time-dependent problems, the **PHASE** parameter is used to specify a drive label. The **SOLVERS** command (page 4-153) allows each value of drive label to be associated with a phase angle (steady-state ac) or a drive function (transient).

The fields from some conductors are calculated using an adaptive integration method which requires the user to supply a **TOLERANCE**. This specifies the error tolerance on the flux density in the units system being used. The field from conductors without a **TOLERANCE** parameter, or with **TOLERANCE** set equal to zero is calculated to a tolerance of 10 gauss.

### *Single Filament Conductors*

A negative value of **TOLERANCE** can be used to request a single filament approximation to the conductors. The absolute value of the **TOLERANCE** is then used to control numerical integration along the trajectory of the current filament (fields are evaluated just by a line integral with no integration over the cross-section).

## Local Coordinate System 1

The local coordinate system 1 is formed by displacing the origin with respect to the global origin to coordinates (**XCEN**, **YCEN**, **ZCEN**) and rotating by Euler angles (**PHI1**, **THETA1**, **PSI1**) (see “Euler Angles” on page 2-31 and Figure 4.5). **ANGLE** is an alias for **PHI1** (see Figure 4.4).

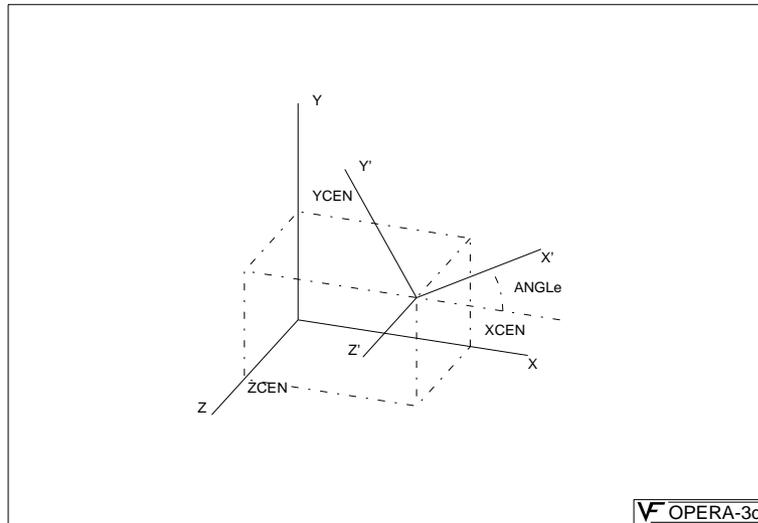


Figure 4.4 Local Coordinate System 1

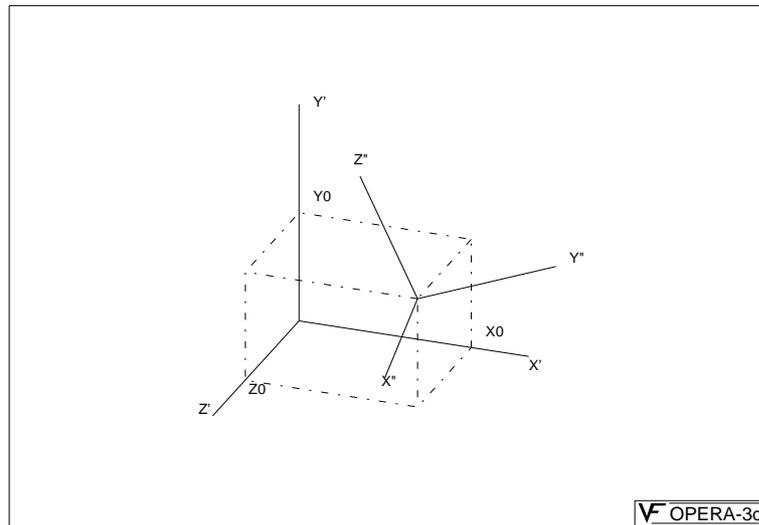
## Local Coordinate System 2

The local coordinate system 2 is formed by displacing the origin with respect to local coordinate system 1 by (**X0**, **Y0**, **Z0**), and rotating by Euler angles (**T**, **P**, **S**) (see “Euler Angles” on page 2-31 and Figure 4.5).

## Symmetries and Reflections

**SOLENOIDS**, **RACETRACKS** and **BEDSTEADS** can be reflected in the global ZX plane, with the currents in the reflection flowing in the same sense around the local Y' axis (**SYMM=2** or **-2**). Any other values of **SYMMETRY** give no reflection.

All other conductors can have more general symmetries. **SYMMETRY=N** gives **N** copies of the basic shape each rotated by  $360/n$  degrees about the global Z axis with respect to the previous copy. If **N** is negative then the sign of the current alternates from one copy to the next.



**Figure 4.5 Local Coordinate System 2**

The other conductors can also have reflections in local(1) coordinate planes. The parameters **IRXY**, **IRYZ** and **IRZX** can have values **0**, **1** or **-1** giving no reflection, reflection with the same sign of current or reflection with reverse sign of current.

## Solenoids

Two types of solenoid are available (Figure 4.6). The first (**SOLENOID**) has a restricted set of symmetries and can only be orientated with local coordinate system 1 without the Euler angles (**PHI1**, **THETA1**, **PSI1**). The second (**GSOLENOID**) can be generally orientated and replicated with any of the symmetries and reflections.

The quadrilateral cross section of the solenoids in the local XY plane is defined by the coordinates of the 4 corners. (**X1**, **Y1**, ..., **X4**, **Y4**) and the curvatures of the sides (**CU1**, ..., **CU4**). If the 4 corners are entered in a clockwise sense when viewed from the positive Z axis, then positive curvature gives an increase in the area of the cross section.

Positive currents flow in the positive Z direction across the positive X half of the XY plane, assuming that the vertices of the cross section have been defined in a clock-wise sense; otherwise the direction of current flow is reversed.

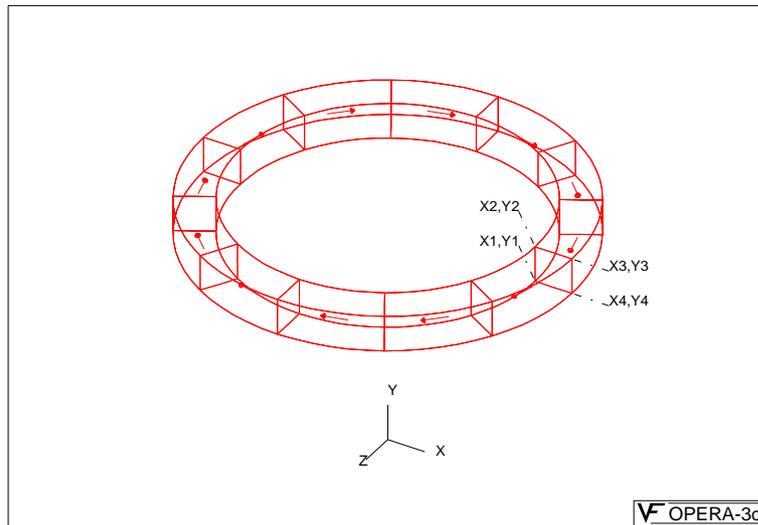


Figure 4.6 A Solenoid

*Parameters for Solenoids*

<b>SOLENOID</b>	
<b>XCENTRE, YCENTRE, ZCENTRE</b>	Local coordinate system 1
<b>X1, Y1, X2, Y2</b>	Solenoid cross section in X'Y' plane
<b>X3, Y3, X4, Y4</b>	Solenoid cross section in X'Y' plane
<b>CU1, CU2, CU3, CU4</b>	Solenoid cross section curvatures
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label

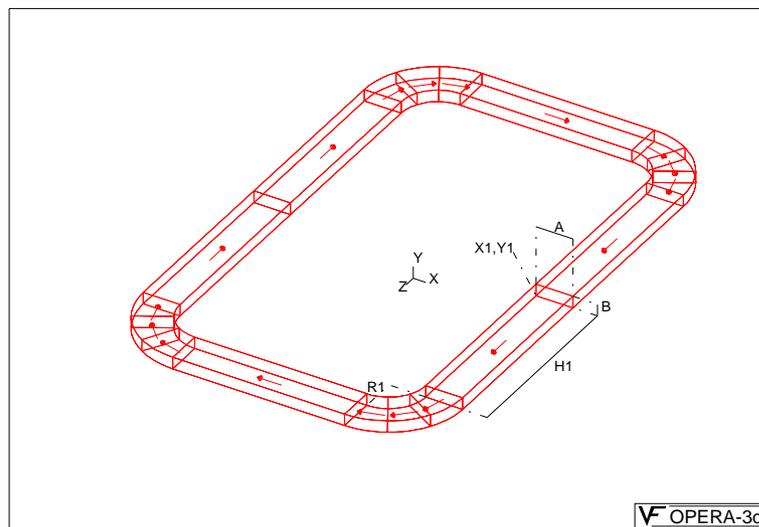
*Parameters for General Solenoids*

<b>GSOLENOID</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>X1, Y1, X2, Y2</b>	Solenoid cross section in X'Y' plane

<b>GSOLENOID</b> (continued)	
<b>X3, Y3, X4, Y4</b>	Solenoid cross section in X'Y' plane
<b>CU1, CU2, CU3, CU4</b>	Solenoid cross section curvatures
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes
<b>TOLE</b>	Field tolerance

## Racetracks

Two types of racetrack are available (Figure 4.7). The first (**RACETRACK**) has



**Figure 4.7 A Racetrack**

the restricted set of symmetries and can only be orientated with local coordinate system 1 without the Euler angles (**PHI1, THETA1, PSI1**). The second (**GRACETRACK**) can be generally orientated and replicated with any of the symmetries and reflections.

The racetrack is made up of 4 straight sections and four 90 degree arcs. The cross section is rectangular, defined by its width in local X direction (**A**) and thickness in local Y direction (**B**). The coordinates of the bottom inside edge as it crosses the XY plane are given by **X1** and **Y1**. The half-length of the z-directed straight is **H1** and the inside radius of the corners is **R1**. **R1** must be greater than zero.

Positive currents flow in the positive Z direction across the positive X half of the XY plane (local coordinates).

### *Parameters for Racetracks*

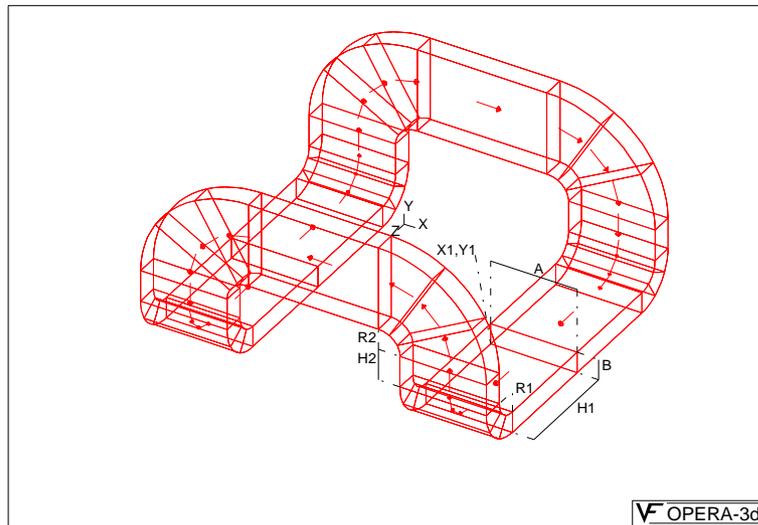
<b>RACETRACK</b>	
<b>XCENTRE, YCENTRE, ZCENTRE</b>	Local coordinate system 1
<b>A, B</b>	Conductor cross section
<b>X1, Y1</b>	Local coordinates of bottom inside corner
<b>H1, R1</b>	Half length and corner radius
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label

### *Parameters for General Racetracks*

<b>GRACETRACK</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>A, B</b>	Conductor cross section
<b>X1, Y1</b>	Local coordinates of bottom inside corner
<b>H1, R1</b>	Half length and corner radius
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes
<b>TOLERANCE</b>	Field tolerance

## **Bedsteads**

Two types of bedstead are available (Figure 4.8). The first (**BEDSTEAD**) has the restricted set of symmetries and can only be orientated with local coordinate system 1 without the Euler angles (**PHI1, THETA1, PSI1**). The second (**GBEDSTEAD**) can be generally orientated and replicated with any of the symmetries and reflections.



**Figure 4.8 A Bedstead**

The bedstead is made up of 8 straight sections and eight 90 degree arcs. The cross section is rectangular, defined by its width in local X direction (**A**) and thickness in local Y direction (**B**). The coordinates of the bottom inside edge as it crosses the XY plane are given by **X1** and **Y1**. The half-length of the z-directed straight is **H1** and the inside radius of the corners at the ends of the z-directed straights is **R1**. The length of the straights in the local y direction is **H2** and the inside radius of the arcs connecting these straights to the straight bars which cross the YZ plane is **R2**. **R1** and **R2** must be greater than zero.

Positive currents flow in the positive Z direction across the positive X half of the XY plane (local coordinates).

### *Parameters for Bedsteads*

<b>BEDSTEAD</b>	
<b>XCENTRE, YCENTRE, ZCENTRE</b>	Local coordinate system 1
<b>A, B</b>	Conductor cross section
<b>X1, Y1</b>	Local coordinates of bottom inside corner
<b>H1, H2</b>	Lengths of straight sections
<b>R1, R2</b>	Inside radii
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label

*Parameters for  
General  
Bedsteads*

<b>BEDSTEAD</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>A, B</b>	Conductor cross section
<b>X1, Y1</b>	Local coordinates of bottom inside corner
<b>H1, H2</b>	Lengths of straight sections
<b>R1, R2</b>	Inside radii
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes
<b>TOLERANCE</b>	Field tolerance

## Helical Ends

The helical end (**HELIX**, Figure 4.9) is the first of two conductors which are wound on the surface of a cylinder. It is made up of two straight sections parallel to the axis of the cylinder (the local Z axis). The half length of the central filament of the straights is **H1**. The cross section is rectangular, defined by the thickness in the radial direction (**A**) and width in the azimuthal direction (**B**). The azimuthal position of the straights on the cylinder is defined by the angle from the mid i.e ZX, plane (**ALPHA**).

Each end of the conductor consists of two helices which extend to azimuthal angle **BETA** measured from the mid-plane and Z coordinate **H2**. The width can change along the helices so that the arc joining the helices together has a width **R2**. **R2** is normally set the same as **B**.

Positive currents flow in the positive Z direction across the positive X half of the XY plane (local coordinates).

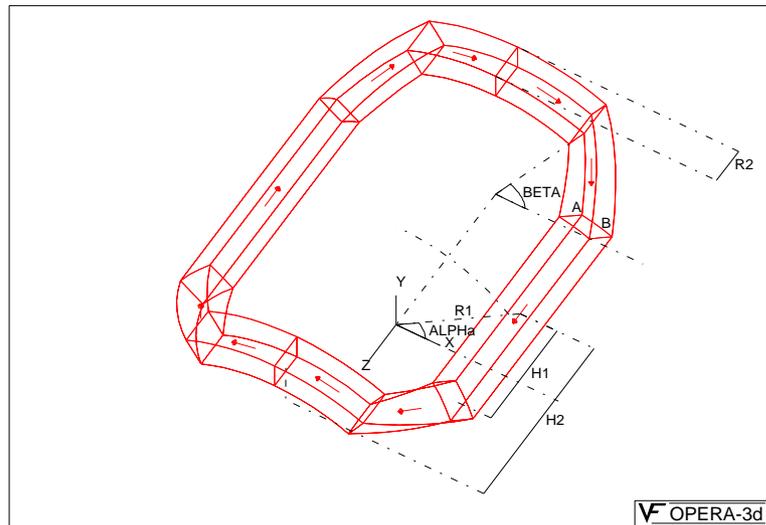


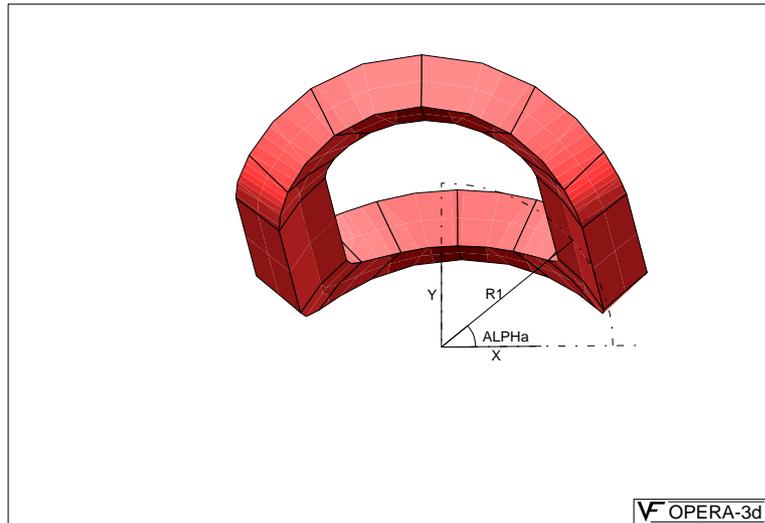
Figure 4.9 A Helical End

### Parameters for Helical Ends

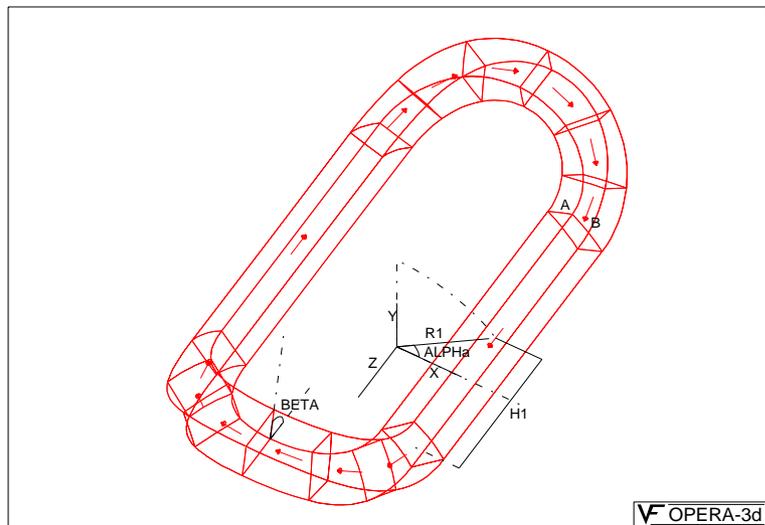
<b>HELIX</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>A, B</b>	Conductor cross section
<b>H1, H2</b>	Half lengths of conductor
<b>R1, R2</b>	Radius of forming cylinder; thickness of cross-over arc
<b>ALPHA, BETA</b>	Angles to central filament of straight and end of helix
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes
<b>TOLERANCE</b>	Field tolerance

## Constant Perimeter Ends

The constant perimeter end (**CPEND**, Figures 4.10 and 4.11) is also wound on the



**Figure 4.10** A Constant Perimeter End



**Figure 4.11** Another view of a Constant Perimeter End

surface of a cylinder. It is made up of two straight sections parallel to the axis of the cylinder (the local Z axis). The half length of the central filament of the straights is **H1**. The cross section is rectangular, defined by the thickness in the radial direction (**A**) and width in the azimuthal direction (**B**). The surface which touches the cylinder can be flat and **TANGENTIAL** to the cylinder or curved and

**FITTING** the cylinder. (N.B. The **FITTING** conductors are more accurate in geometry and field than the **TANGENTIAL**.) The azimuthal position of the straights on the cylinder is defined by the angle from the mid. i.e ZX, plane (**ALPHA**).

The ends of the conductor form a smooth curve over the cylinder. In manufacture they are machined by a cutter which has an angle **BETA** to the local Z axis of the cylinder and traverses a circular path on a cylinder of radius **R2**.

Positive currents flow in the positive Z direction across the positive X half of the XY plane (local coordinates).

*Parameters for  
Constant  
Perimeter Ends*

<b>CPEND</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>A, B</b>	Conductor cross section
<b>ALPHA, BETA</b>	Angles to central filament of straight and of cutting tool
<b>H1</b>	Half length of straight
<b>R1, R2</b>	Radii of construction cylinders
<b>FIT</b>	Conductor fit to mandrel
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes
<b>TOLERANCE</b>	Field tolerance

## Straight Bars

The **STRAIGHT** bar, Figure 4.12 is a simple rectangular cross section conductor. Its cross section is **A** in the local X direction and **B** in the local Y direction. Its central filament starts at the local origin and extends **H1** in the local Z direction. Positive currents flow in the positive local Z direction.

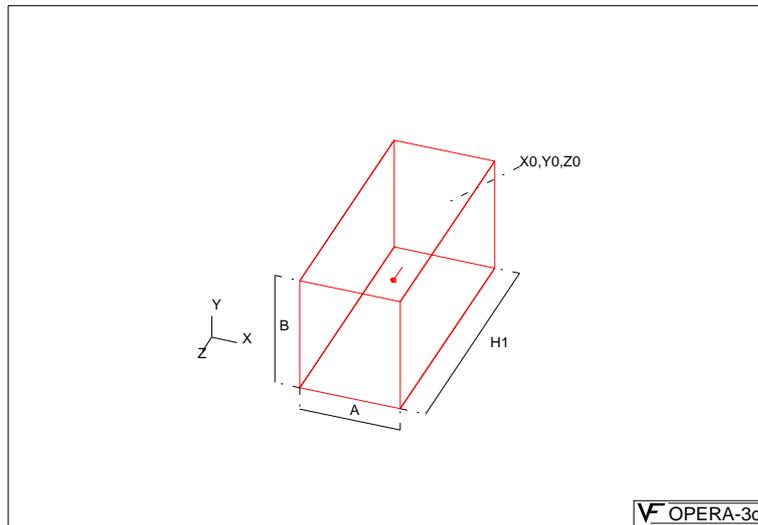


Figure 4.12 A Straight Bar

### Parameters for Straight Bars

<b>STRAIGHT</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>A, B</b>	Conductor cross section
<b>H1</b>	Length of straight
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes

### Circular Arcs

The circular **ARC**, Figure 4.13 is similar to the straight bar. Its cross section is **A** in the local X direction and **B** in the local Y direction. Its central filament starts at the local origin and moves in the local Z direction. It bends through an angle **PHI** towards the positive local Y direction. Positive currents flow from the starting coordinates in the positive local Z direction.

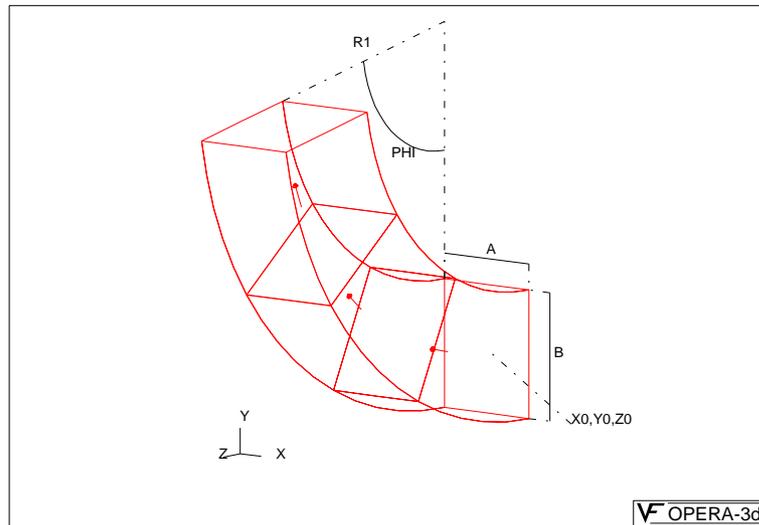


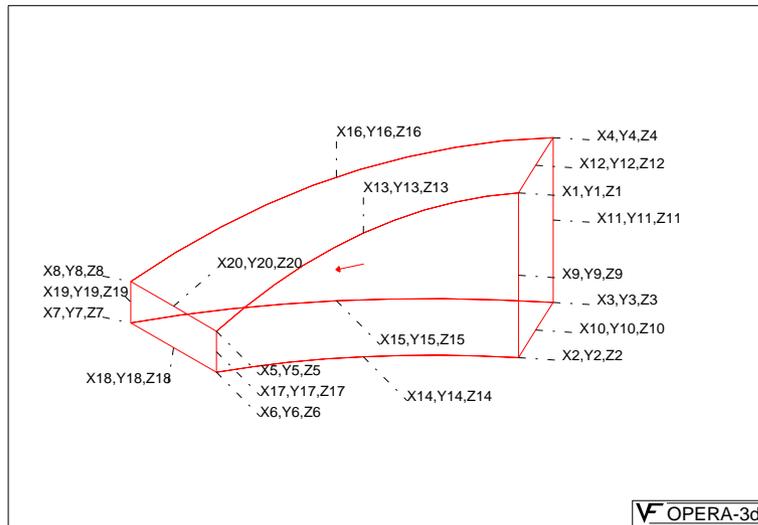
Figure 4.13 An Arc

### Parameters for Circular Arcs

<b>ARC</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>A, B</b>	Conductor cross section
<b>R1, PHI</b>	Inside radius and angle
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes

### Bricks

Two brick elements can be used to defined more complex shapes, especially those which involve changes in cross section. They have 8 nodes (**BR8**) or 20 nodes (**BR20**, Figure 4.14).



**Figure 4.14 A 20-Node Brick**

The bricks are defined by the coordinates of the nodes ( $X_1, Y_1, Z_1, \dots, X_{20}, Y_{20}, Z_{20}$ ). The current in a brick flows from face 1 (nodes 1, 2, 3, 4) to face 2 (nodes 5, 6, 7, 8). The 20 noded brick has mid-edge nodes:

node 9	between nodes 1 and 2
node 10	between nodes 2 and 3
node 11	between nodes 3 and 4
node 12	between nodes 4 and 1
node 13	between nodes 1 and 5
node 14	between nodes 2 and 6
node 15	between nodes 3 and 7
node 16	between nodes 4 and 8
node 17	between nodes 5 and 6
node 18	between nodes 6 and 7
node 19	between nodes 7 and 8
node 20	between nodes 8 and 5

If the mid-edge nodes are not co-linear with the corner nodes, then the surfaces are parabolic. If the mid-edge nodes are not at the centre of the edges, the current density will vary over the cross section. The mid-edge nodes should not be displaced outside the  $\frac{1}{4}$  and  $\frac{3}{4}$  points.

The top-level **DEFINE** command of the pre processor (page 4-49) can be used to define the geometry of brick conductors. This allows triangular or quadratic cross sections, which may be curved, to be extruded in the third direction. When the

extrusions are complete the geometry is copied into the conductor storage area so that other conductor parameters, local coordinate systems, replications, current density etc. can be set to their correct values with the **MODIFY** sub-command.

### *Parameters for 8-Noded Bricks*

<b>BR8</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>X1, Y1, Z1</b>	Bottom right corner of front face
<b>X2, Y2, Z2</b>	Top right corner of front face
<b>X3, Y3, Z3</b>	Top left corner of front face
<b>X4, Y4, Z4</b>	Bottom left corner of front face
<b>X5, Y5, Z5</b>	Bottom right corner of back face
<b>X6, Y6, Z6</b>	Top right corner of back face
<b>X7, Y7, Z7</b>	Top left corner of back face
<b>X8, Y8, Z8</b>	Bottom left corner of back face
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes
<b>TOLERANCE</b>	Field tolerance

### *Parameters for 20-Noded Bricks*

<b>BR20</b>	
<b>XCENTRE, YCENTRE, ZCENTRE, PHI1, THETA1, PSI1</b>	Local coordinate system 1
<b>X0, Y0, Z0</b>	Local coordinate system 2 (origin)
<b>T, P, S</b>	Local coordinate system 2 (Euler angles)
<b>X1, Y1, Z1</b>	Bottom right corner of front face
<b>X2, Y2, Z2</b>	Top right corner of front face
<b>X3, Y3, Z3</b>	Top left corner of front face
<b>X4, Y4, Z4</b>	Bottom left corner of front face

<b>BR20</b> (continued)	
<b>X5, Y5, Z5</b>	Bottom right corner of back face
<b>X6, Y6, Z6</b>	Top right corner of back face
<b>X7, Y7, Z7</b>	Top left corner of back face
<b>X8, Y8, Z8</b>	Bottom left corner of back face
<b>X9, Y9, Z9</b>	Mid-point of right edge of front face
<b>X10, Y10, Z10</b>	Mid-point of top edge of front face
<b>X11, Y11, Z11</b>	Mid-point of left edge of front face
<b>X12, Y12, Z12</b>	Mid-point of bottom edge of front face
<b>X13, Y13, Z13</b>	Mid-point of bottom right edge
<b>X14, Y14, Z14</b>	Mid-point of top right edge
<b>X15, Y15, Z15</b>	Mid-point of top left edge
<b>X16, Y16, Z16</b>	Mid-point of bottom left edge
<b>X17, Y17, Z17</b>	Mid-point of right edge of back face
<b>X18, Y18, Z18</b>	Mid-point of top edge of back face
<b>X19, Y19, Z19</b>	Mid-point of left edge of back face
<b>X20, Y20, Z20</b>	Mid-point of bottom edge of back face
<b>CURD, SYMMETRY, PHASE</b>	Current density, symmetry and drive label
<b>IRXY, IRYZ, IRZX</b>	Reflections in local coordinate system 1 coordinate planes
<b>TOLERANCE</b>	Field tolerance

## The CONDUCTOR Sub-command DEFINE

### Menu Route:

DEFINE↓

Conductors → Define a conductor

### Command Line Parameters:

Sub-command	DEFINE		
Parameter	Default	Function	
END	none	Conductor type	
		SOLENOID	Simple solenoid
		GSOLENOID	Generally orientated solenoid
		RACETRACK	Simple racetrack
		GRACETRACK	Generally orientated racetrack
		BEDSTEAD	Simple bedstead
		GBEDSTEAD	Generally orientated bedstead
		HELIX	Helical end racetrack
		CPEND	Constant perimeter end racetrack
		STRAIGHT	Straight bar
		ARC	Circular arc
		BR8	8-noded brick
		BR20	20-noded brick
DEFAULT	max	Number of an existing conductor used to reference default values. If no number is given, then the highest numbered conductor of the same type is used.	

The program prompts the user to provide values for all the parameters appropriate for the conductor type. The input lines are decoded in the same way as top-level commands without a command name. This means that the help character, !, can be used to get more information about the parameters and can be used twice, !!, to get individual parameter prompting.

The default value is given in parentheses following each parameter name. If no value is given, there is no default and a value must be supplied.

The full parameter lists were given above with the descriptions of the conductors, in the order in which they are requested from the user.

### Examples

- Example: In the example which follows various forms of input are used during the definition of a solenoid.

```

OPERA > cond
Define and list conductor data
Conductor definition and editing.

Sub-commands: Define, Modify, Erase, Print, Write and
Quit; ! for help.
COND > define solenoid
Define conductors.
Specify new values by:
giving values positionally
OR assigning PARAMETER=value
OR hitting <return> to accept all defaults.

Conductor 1: Solenoid around Y' axis.
Local coordinate system 1
XCENTRE(0), YCENTRE(0), ZCENTRE(0)
COND > xcen=5 ycen=3 zcen=2
Solenoid cross section in X'Y' plane.
X1(), Y1(), X2(), Y2()
COND > !
Parameter Value Meaning
X1           Local X coordinate of bottom
             inside corner
Y1           Local Y coordinate of bottom
             inside corner
X2           Local X coordinate of top inside
             corner
Y2           Local Y coordinate of top inside
             corner

COND > 3 3 3 5
Solenoid cross section in X'Y' plane.
X3(), Y3(), X4(), Y4()
COND > 5 5 5 3
Solenoid cross section curvatures.
CU1(0), CU2(0), CU3(0), CU4(0)
COND > cu3=0.05
Current density, symmetry and drive label
CURD(), SYMMETRY(1), PHASE(ONE)
COND > !!
There are 3 parameters:
For each parameter:
hit return to accept default
OR enter new value
OR type $H for help

```

OR type \$S to skip remaining parameters and execute command

OR type \$A to skip remaining parameters and abort command

```

NO.   Name      Value  Meaning
1     CURD                      Current density
!! > 10
2     SYMMETRY 1      Symmetry code (1 or 2)
!! > 2
3     PHASE     ONE    Drive label
!! >
Type return to obey command or $A to abort
!! >

```

Sub-commands: Define, Modify, Erase, Print, Write and Quit; ! for help.

COND >

## The CONDUCTOR Sub-command ERASE

### Menu Route:

```

DEFINE↓
  Conductors → Erase conductors

```

### Command Line Parameters:

Sub-command	ERASE	
Parameter	Default	Function
<b>C1</b>	<i>none</i>	Number of first conductor to be erased.
<b>C2</b>	<i>none</i>	Number of last conductor to be erased. <b>C2</b> can be set to * to indicate the highest numbered conductor.

The **ERASE** sub-command removes a range of conductors from the conductor database. For example, if **C1** is set to **3** and **C2** to **6** then conductors with numbers **3, 4, 5** and **6** will be erased. **C2** can be set to **\*** to indicate the highest numbered conductor.

The remaining conductors with numbers greater than **C2** are renumbered to give a contiguous set.

## The CONDUCTOR Sub-command MODIFY

### Menu Routes:

DEFINE↓  
Conductors → Modify conductors

MODIFY↓  
Conductors

### Command Line Parameters:

Sub-command	MODIFY	
Parameter	Default	Function
<b>C1</b>	<i>none</i>	Number of first conductor to be modified.
<b>C2</b>	<i>none</i>	Number of last conductor to be modified. <b>C2</b> can be set to * to indicate the highest numbered conductor.
<b>END</b>	<i>none</i>	Conductor type: <b>SOLENOID, GSOLENOID, RACETRACK, GRACETRACK, BEDSTEAD, GBEDSTEAD, HELIX, CPEND, STRAIGHT, ARC, BR8 or BR20.</b>
<b>SYMMETRY</b>	<i>none</i>	Symmetry code.
<b>XCENTRE</b>	<i>none</i>	X coordinate of origin of local system 1.
<b>YCENTRE</b>	<i>none</i>	Y coordinate of origin of local system 1.
<b>ZCENTRE</b>	<i>none</i>	Z coordinate of origin of local system 1.
<b>CURD</b>	<i>none</i>	Current density.
<b>PHI1</b>	<i>none</i>	Euler angle phi (local system 1).
<b>TOLERANCE</b>	<i>none</i>	Tolerance on fields.
<b>X0</b>	<i>none</i>	X coordinate of origin of local system 2.
<b>Y0</b>	<i>none</i>	Y coordinate of origin of local system 2.
<b>Z0</b>	<i>none</i>	Z coordinate of origin of local system 2.
<b>T</b>	<i>none</i>	Euler angle theta (local system 2).
<b>P</b>	<i>none</i>	Euler angle phi (local system 2).
<b>S</b>	<i>none</i>	Euler angle psi (local system 2).
<b>IRXY</b>	<i>none</i>	Reflection code in xy plane of local system 1.
<b>IRYZ</b>	<i>none</i>	Reflection code in yz plane of local system 1.
<b>IRZX</b>	<i>none</i>	Reflection code in zx plane of local system 1.
<b>X1</b>	<i>none</i>	X coordinate of corner of conductor cross section.

Sub-command	<b>MODIFY</b> (continued)	
Parameter	Default	Function
<b>Y1</b>	<i>none</i>	Y coordinate of corner of conductor cross section.
<b>X2</b>	<i>none</i>	X coordinate of corner of solenoid cross section.
<b>Y2</b>	<i>none</i>	Y coordinate of corner of solenoid cross section.
<b>X3</b>	<i>none</i>	X coordinate of corner of solenoid cross section.
<b>Y3</b>	<i>none</i>	Y coordinate of corner of solenoid cross section.
<b>X4</b>	<i>none</i>	X coordinate of corner of solenoid cross section.
<b>Y4</b>	<i>none</i>	Y coordinate of corner of solenoid cross section.
<b>A</b>	<i>none</i>	Thickness of conductor in x or radial direction.
<b>B</b>	<i>none</i>	Width of conductor in y or azimuthal direction.
<b>H1</b>	<i>none</i>	Length of straight section.
<b>H2</b>	<i>none</i>	Length of upright ( <b>BEDSTEAD</b> )
		Local Z coordinate of midpoint of cross-over ( <b>HELIX</b> ).
<b>R1</b>	<i>none</i>	Radius: inner radius of arc ( <b>RACETRACK</b> , <b>BEDSTEAD</b> , <b>ARC</b> ).
		Radius of cylinder ( <b>HELIX</b> , <b>CPEND</b> ).
<b>R2</b>	<i>none</i>	Radius: inner radius of arc ( <b>BEDSTEAD</b> ).
		Width of cross-over ( <b>HELIX</b> ).
		Radius of generating cylinder ( <b>CPEND</b> ).
<b>PHI</b>	<i>none</i>	Angle of ARC.
<b>ALPHA</b>	<i>none</i>	Angle of straight from mid plane of cylinder ( <b>HELIX</b> and <b>CPEND</b> ).
<b>BETA</b>	<i>none</i>	Angle of end of helix ( <b>HELIX</b> ), or cutter ( <b>CPEND</b> ).
<b>CU1</b>	<i>none</i>	Curvature of cross section of solenoid (points 1 to 2).
<b>CU2</b>	<i>none</i>	Curvature of cross section of solenoid (points 2 to 3).
<b>CU3</b>	<i>none</i>	Curvature of cross section of solenoid (points 3 to 4).
<b>CU4</b>	<i>none</i>	Curvature of cross section of solenoid (points 4 to 1).
<b>FIT</b>	<i>none</i>	Fit of straight section to cylinder: <b>TANGENTIAL</b> or <b>FITTING</b> ( <b>CPEND</b> ).
<b>PHASE</b>	<i>none</i>	Drive label.
<b>THETA1</b>	<i>none</i>	Euler angle theta (local system 1).
<b>PSI1</b>	<i>none</i>	Euler angle psi (local system 1).

The **MODIFY** command can be used in two ways. If only the range of conductors is specified on the **MODIFY** sub-command then the program issues the same prompts as are used by the **DEFINE** sub-command (page 4-40). The displayed values are taken from the first conductor in the range and any changes made by the user are applied to all the conductors within the range which are of the same type.

The second way of using **MODIFY** makes use of the other parameters of the **MODIFY** command. The values of the parameters used on the command are applied to all the conductors in the range. This way of using **MODIFY** should be used with care, especially if more than one conductor is specified by **C1** and **C2**, and the range includes conductors of more than one type. It is possible to create conductors with invalid data, since some of the parameters have different meanings for different conductor types. However it is very powerful for changing parameters which apply to all conductors, such as the current density (**CURD**).

**N.B.** Only the second way of using **MODIFY** is available from the GUI.

Expressions can also be used to good effect in this second way of using **MODIFY**. The expressions are evaluated for each conductor; within each evaluation any parameters which are referenced have the correct values for the conductor concerned. However, the original values are used in each expression. Thus setting **A=A\*2**, **CURD=CURD/A/B** would set the current density using the original value of **A**, not the updated value.

- Example: **MODIFY**ing the conductor defined above.

```
Sub-commands: Define, Modify, Erase, Print, Write and
Quit; ! for help.
COND > modi 1 1
Modify conductors.
Modifying conductor 1.
```

```
Specify new values by:
giving values positionally
OR assigning PARAMETER=value
OR hitting <return> to accept all defaults.
```

```
Conductor 1: Solenoid around Y' axis.
Local coordinate system 1
XCENTRE(5), YCENTRE(3), ZCENTRE(2)
COND >
Solenoid cross section in X'Y' plane.
X1(3), Y1(3), X2(3), Y2(5)
COND > x1=2 x2=2
```

```

Solenoid cross section in X'Y' plane.
X3(5), Y3(5), X4(5), Y4(3)
COND > x3=4,,4
Solenoid cross section curvatures.
CU1(0), CU2(0), CU3(0.05000), CU4(0)
COND >
Current density, symmetry and drive label
CURD(10), SYMMETRY(1), PHASE(ONE)
COND >

```

Sub-commands: Define, Modify, Erase, Print, Write and Quit; ! for help.

```

COND > modi 1 * curd=curd/2
Modify conductors.

```

## The CONDUCTOR Sub-command PRINT

### Menu Route:

```

DEFINE↓
  Conductors → Print data

```

### Command Line Parameters:

Sub-command	PRINT	
Parameter	Default	Function
<b>C1</b>	1	Number of first conductor to be printed.
<b>C2</b>	*	Number of last conductor to be printed. <b>C2</b> can be set to * to indicate the highest numbered conductor.

The **PRINT** sub-command lists the parameters of the selected range of conductors to the terminal and to the log file. It uses the same format for the prompts in **DEFINE** and **MODIFY**.

## The CONDUCTOR Sub-command QUIT

**Menu Route:** DEFINE↓  
Conductors → Return

**Command Line Parameters:**

Sub-command	QUIT
No Parameters	

The **QUIT** sub-command leaves the **CONDUCTOR** command and returns to the top-level commands.

## The CONDUCTOR Sub-command WRITE

**Menu Route:** DEFINE↓  
Conductors → Write data to a file

**Command Line Parameters:**

Sub-command	WRITE	
Parameter	Default	Function
FILE	<i>none</i>	Name of file.

The **WRITE** sub-command creates a command input **FILE** containing the **CONDUCTOR** command, the set of **DEFINE** sub-commands for all the conductors currently defined and the **QUIT** sub-command.

In the pre processor, conductor files are useful for copying the conductors from one data set to another, or for transferring the data into the post processor. The actual commands used to define the conductors are also included in any file created by the pre processor top level command **WRITE**.

If no file name extension is given, the program adds the extension *cond*.

*cond* files can be read into the programs using the **READ** command (page 4-149) or the 'built-in' command, **\$ COMINPUT** (page 2-22).

## The **DEFINE** Command

---

### Menu Route:

```

DEFINE↓
  Define new mesh → Finite element mesh
                  → 8 or 20 node conductors
  
```

### Command Line Parameters:

Command	<b>DEFINE</b>			
Parameter	Default	Function		
<b>TYPE</b>	<b>MESH</b>	Type of elements to be defined.		
		<table border="0"> <tr> <td><b>CONDUCTOR</b></td> <td>8 or 20 node conductor elements</td> </tr> <tr> <td><b>MESH</b></td> <td>Finite elements mesh</td> </tr> </table>	<b>CONDUCTOR</b>	8 or 20 node conductor elements
<b>CONDUCTOR</b>	8 or 20 node conductor elements			
<b>MESH</b>	Finite elements mesh			
<b>KEEP</b>	<b>YES</b>	Keep existing construction lines		
		<b>NO</b>	Construction lines removed at start of <b>DEFINE</b> sequence.	
		<b>YES</b>	Construction lines kept from any previous <b>DEFINE</b> commands.	
<b>THREED</b>	<b>NO</b>	Use 3d Viewer		
		<b>NO</b>	No 3d Viewer	
		<b>YES</b>	3d View starts when first layer is complete.	

**DEFINE** is the command which puts the OPERA-3d pre processor into its finite element mesh creation mode. The pre processor only supports one type of finite element mesh construction: a surface defined by a set of facets is specified and is then extruded or swept through space to create a volume discretisation. The user input is structured by the program. Points used to define the geometry are defined first, followed by the surface facets and the element subdivision of these facets. It is possible to define more points while facets are being defined, and more facets while subdivisions are being defined. The surface of facets is then extruded or swept through space thus creating a series of layers of volumes. The topology of the initial surface is maintained during the extrusion operations, but the point coordinates can be changed in the new surface created by each extrusion or sweep. Once the complete model has been created, the volumes in the layers of the mesh can be assigned material attributes and boundary conditions can be assigned to the facets of the volumes.

The **TYPE** parameter allows definition of the finite element mesh for the solution of the model (**TYPE=MESH**) or 8 or 20 node brick conductor elements

(**TYPE=CONDUCTOR**). After definition conductor elements are copied into the conductor storage area as **BR8** or **BR20** conductors with total current of 1 Amp and can only be modified by use of the **CONDUCTOR** sub-commands. (N.B. The cross sections of conductors can be only triangles or quadrilaterals.) The finite element mesh can be modified using **MODIFY** and have extra layers added using **EXTEND**.

The **KEEP** parameter affects construction lines defined during a previous use of the **DEFINE** command. If **KEEP=NO**, any construction lines which already exist are removed before any new data is defined.

The **THREED** parameter can be used to request the 3d Viewer to be used during the **DEFINE** command to display the outline of the model. When **THREED=YES** is selected, the viewer is started when the first layer has been created and is updated whenever the two dimensional display is refreshed.

Labels are assigned to each entity (i.e. point, line, facet or volume) in the mesh. All entities have the label **ALL**, facets have boundary condition names and volumes have material names, potential types and element types. Additional labels can be added or removed with the **LABEL** command (page 4-125). These labels can be used to select subsets of the entities for **DISPLAY**.

Each time the **DEFINE** command is used to create a finite element mesh, a new mesh is started. Up to 100 meshes can be defined, each one forming part of any data set prepared for analysis. The user is responsible for ensuring that the meshes are contiguous at their interfaces.

## Finite Element Meshing

The elements used by the analysis programs are created by subdividing volumes created by the pre processor. These volumes are formed by extruding facets on the base plane through space. There are two types of mesh available.

- If all the facets are made up from 3 or 4 lines with regular subdivisions, the volumes can be meshed using hexahedra (which can be degenerate). Regular subdivision means that for 4 sided facets, opposite sides must have the same subdivision and for 3 sided facets, two sides must have the same subdivision.
- Any model can be meshed with tetrahedra. Tetrahedral meshes can be generated in extruded polygonal volumes with no restrictions on the subdivision of the edges. The points defining any polygonal facet must be coplanar.

In either type of mesh, the lines can be straight (2 points) or quadratic curves (3 points). The position of the mid-side point of quadratic curves affects the subdivision of the volumes. If it is not central there will be smaller elements at the end

of the side to which it is closer. Any combination of straight and curved lines may be used. Thus, facets can be anything from 3 noded triangles to many-noded curved sided polygons. In the same way extrusions can be linear or quadratic. Quadratic extrusions have mid-side points which can be moved independently, allowing definition of curves or non-uniform subdivision in the extrusion direction.

## DEFINE start up sequence and modes

### Coordinate system

After the **DEFINE** command has been issued, the user will be asked to select the default coordinate system to be used for input and display of the initial surface (base plane). There are 3 pre-defined systems

<b>Pre-defined Coordinate Systems</b>		
Keyword	Menu item	Coordinate system
<b>XY</b>	XY plane, extrude in Z	Same as global, extrusion direction Z.
<b>YZ</b>	YZ plane, extrude in X	U=Y, V=Z, W=X, extrusion direction X.
<b>ZX</b>	ZX plane, extrude in Y	U=Z, V=X, W=Y, extrusion direction Y.
<b>NEW</b>	(keyboard only)	Define a new coordinate system.

However any other right handed system can be defined to give the orientation of the base plane and the default extrusion direction. The coordinate system is defined by its origin, and Euler angles ([page 2-31](#)). (This coordinate system can be changed during **DEFINE** using **Change View** menu item or the **V** cursor hit.) The most appropriate system should be selected. The default extrusions will be normal to the plane selected and unless a coordinate triple is specified for a point, the point will be given the default coordinate of the plane. Once a plane has been selected, subsequent input of point coordinates is specified in a local U, V, W coordinate system, where U and V are in the plane and W is out of the plane.

### W coordinate

The user is next asked to input the default W coordinate of the plane, for points whose W coordinate is not explicitly defined.

### Window size

Finally, the user is asked to input window size to be used for subsequent graphical display of the point and surface data. This is only an initial specification and can be changed at any time subsequently. The input request is for **umin**, **umax**, **vmin**

and *vmax* which may be supplied in free format. The screen is cleared at this point and the program enters Point Definition Mode.

The graphics display indicates, at the top, the mesh number and the plane or extrusion layer number currently being worked on, and, at the bottom, the definition or name of the local (UVW) coordinate system.

## Aspect Ratio Searching

When the user selects a point or line on the screen, the program searches for the point or line nearest to the cursor position in real coordinate space. If the horizontal and vertical axes limits are very different, i.e. if

$$\alpha = \frac{u_{\max} - u_{\min}}{v_{\max} - v_{\min}} \quad (4.1)$$

and  $\alpha \gg 1$  or  $\alpha \ll 1$  it might be difficult to select the intended object.

To make it possible to select correctly in such circumstances, aspect ratio searching can be switched on or off using cursor command **Z** or menu item **Aspect ratio search**. When it is switched on, the program makes use of the window aspect ratio ( $\alpha$ ), to find the nearest object.

## Defining Meshes with the GUI

When the **DEFINE** command is used with the GUI, the Base Plane Definition Modes are presented on one menu:

Define Baseplane Menu	
Fast Polygon Input	To define points line and facets using grids or construction lines (see <a href="#">page 4-60</a> ).
Point Input	To define points using keyboard input of coordinates, grids and construction lines (see <a href="#">page 4-56</a> ).
Facet Input	To select points to form facets (see <a href="#">page 4-63</a> ).
Subdivision	To subdivide lines (facet sides) to set the finite element mesh size (see <a href="#">page 4-69</a> ).
Extrude	To create and edit the first layer (coordinates, materials and boundary conditions (see <a href="#">page 4-73</a> ).
Escape from baseplane	To create the first layer of size 1 with no editing (if there is at least one facet).

The two input methods, **Fast Polygon Input** and **Point Input** with **Facet Input** can be used separately or together to create the base plane facets.

There are several additional points which should be noted:

- All the data for the first layer should be defined first. The **Define new mesh** option should be used to define the first layer and the **Extend existing mesh** option for one or more additional layers (see “**The EXTEND Command**” on page 4-113).
- Not all options are available from the menus. For example, the coordinate system menu only allows the 3 pre-defined systems.
- It is not possible to abort the definition; the layer must be completed. However, as soon as there is at least one facet, the menu option **Escape from baseplane** is available. This completes all remaining modes of the baseplane definition and extrudes by one unit. The model can be subsequently edited using the **REDEFINE** command (page 4-151).

## Defining Meshes with the Keyboard

In keyboard mode, the program steps through the modes of the **DEFINE** command sequentially, each one being terminated with a **Q** cursor hit. During Point Definition Mode, Fast Polygon Input can be accessed; during Facet Definition Mode, Group Operations can be accessed. There are cursor and text commands which can be used to move between the various modes in almost any order.

## Construction Lines and Grids

Whenever point coordinates can be defined, they can be supplied graphically using construction lines or grids.

### *Construction lines*

Construction lines are straight lines (**LINE**) or arcs (**ARC**). The arc is in fact a linear interpolation in the cylindrical polar coordinate system. It will only be an arc of a circle if the radial coordinates of the end points are the same. The definitions can be given numerically or by choosing points which have already been defined: **LINES** are defined by two **<space>** cursor hits; **ARCS** are defined by either a **C** and two **<space>**s to specify the centre and two end points or by 3 **<space>** cursor hits to specify points on a circle. Cursor defined **LINES** extend beyond the points which define them. Cursor defined **ARCS** are minor arcs if defined by centre and end points or complete circles if defined by 3 points on the circumference.

The menu item **Enter C\_lines** or cursor hit **I** can be used for define construction lines.

- *In keyboard mode* this produces a request to specify data defining construction lines, together with the format of the specification.
- *In menu mode* the type of construction line can be selected from a menu and the values supplied via a ParameterBox:

Construction Line Sub-commands	
Command	Parameters and Function
<b>ARC</b>	<i>uc vc r1 t1 r2 t2</i> An arc centred on ( <i>uc, vc</i> ) starting at polar coordinates( <i>r1, t1</i> ) and ending at ( <i>r2, t2</i> ). Both <i>r</i> and <i>t</i> vary linearly between the end points.
<b>ARC</b>	<b>CURSOR</b> An arc defined by cursor hits: <b>Either:</b> select points for centre with <b>C</b> and end points with <b>&lt;space&gt;</b> . <b>or:</b> select 3 points on circumference of circle with <b>&lt;space&gt;</b> .
<b>LINE</b>	<i>u1 v1 u2 v2 angle</i> A straight line from ( <i>u1, v1</i> ) to ( <i>u2, v2</i> ) rotated by <i>angle</i> around ( <i>u1, v1</i> ).
<b>LINE</b>	<b>CURSOR</b> A straight line through 2 existing points chosen by <b>&lt;space&gt;</b> cursor hits.
<b>QUIT</b>	End the definition of construction lines.

Points can be defined on the nearest construction line using menu item **On nearest C\_line** or cursor hit **N**, or at the nearest intersection of construction lines using menu item **At C\_line intersection** or cursor hit **X**. The set of intersections also includes the end points of the construction lines. In each case the cursor hit only specifies the U and V coordinates. The W coordinate used is the default coordinate of the plane.

The nearest construction line can be erased using menu item **Remove C\_lines** or cursor hit **E** and the lines can be listed at the terminal using menu item **List C\_lines** or cursor hit **L**.

## Grids

Grids are two-dimensional arrays of points in cartesian or polar coordinates. A grid can be switched on or off following menu item **Grid** or **G** cursor hit.

Grid Sub-commands	
Command	Parameters and Function
<b>CARTESIAN</b>	$du\ dv$ Define grid with spacing $du$ in the horizontal direction and $dv$ in the vertical.
<b>POLAR</b>	$dr\ d\theta$ Define grid with spacing $dr$ in the radial direction and $d\theta$ in the azimuthal direction.
<b>NONE</b>	Remove any existing grid.

When a grid exists, and menu item **At the mouse** or cursor hit **<space>** selects the closest grid point.

## Point Definition Mode

The user must specify points to define the corners and mid-side points of the facets in the base plane. Additional points can be entered by returning to Point Definition Mode from Facet Definition Mode. It is also possible to copy points and facets in the Facet Group Operations Mode.

Points are defined using the graphics cursor or numerically using the keyboard. The cursor can be used to position points at the cross-hairs position, or at points on construction lines or grids. Keyboard input can be in cartesian or cylindrical polar coordinates with respect to the current local coordinate system origin.

When enough points have been defined, the user should leave the Point Definition Mode and move on to Facet Definition Mode.

Full details of construction lines and grids are give above and the keyboard input and the cursor commands are described in the following sections.

### Keyboard input

Keyboard input mode can be used to specify coordinates of points in

- cartesian coordinates: menu item **Give U, V, W** or cursor hit **C**
- cylindrical polar coordinates: menu item **Give R, Theta, W** or cursor hit **P**. The origin of the polar coordinate system is the same as the UVW system, with the axis in the W direction and zero azimuthal angle (T) in the U direction.

For each point 1, 2 or 3 values or expressions should be specified. The first defines the U or R coordinate. If it is omitted by using a comma at the start of the command line, the value of U or R will be the last value given for U or R. The second value defines the V or T coordinate. If it is omitted the value of V or T will be the last value used for V or T. When keyboard input mode is entered, U and V have default values taken from the cursor position, and R and T are both zero. The third value defines the W coordinate and always defaults to the default value for the plane.

To return to cursor mode type the keyword **CURSOR** or select the **Quit** button in the DialogBox.

### *The Point Mode Menu and Cursor Hits*

Point Mode Menu and Cursor Hits		
Cursor hit	Menu item	Function
<space>	At mouse	Define a point at the cursor cross hair position or closest grid point.
<b>A</b>		Abort the <b>DEFINE</b> command. If Point Definition Mode has been re-entered from Facet Definition Mode then <b>A</b> acts like <b>Q</b> .
<b>C</b>	Give U, V, W	Switch to input from keyboard in cartesian coordinates UVW.
<b>D</b>	Delete point	Delete point closest to cursor cross hair.
<b>E</b>	Remove C_line	Erase construction line closest to cursor cross hair.
<b>F</b>		Enter Fast Polygon Input Mode (see <a href="#">page 4-60</a> ).
<b>G</b>	Grid	Define or remove a grid (see <a href="#">page 4-55</a> ).
<b>H</b>		Display menu help message explaining all the cursor options.
<b>I</b>	Enter C_lines	Input construction line specifications (see <a href="#">page 4-53</a> ).
<b>L</b>	List C_lines	List construction line specifications.

<b>Point Mode Menu and Cursor Hits (continued)</b>			
Cursor hit	Menu item	Function	
<b>M</b>	Move point	Move the nearest point. The point can be repositioned using cursor hits which have the same meanings as for point definition. See <a href="#">page 4-58</a> .	
<b>N</b>	On nearest C_line	Define a point on the nearest construction line with minimum normal distance.	
<b>P</b>	Give R, Theta, W	Switch to input from keyboard in local cylindrical polar coordinates R $\theta$ W.	
<b>Q</b>	Go to Facet Definition	Leave Point Definition Mode and move on to Facet Definition Mode.	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
<b>&lt;return&gt;</b>	Reconstruct at the same size.		
<b>T</b>	Show coordinates	Type the coordinates of the point closest to the cross-hairs. The coordinates are given in cartesian (UVW) and polar (R $\theta$ W) coordinates.	
<b>U</b>	Undo	Undo the last move.	

<b>Point Mode Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.
<b>X</b>	At C_line intersection	Define a point at the closest construction line intersection or end.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off (page 4-52).

*The Point  
Repositioning  
Mode Menu and  
Cursor Hits*

<b>Point Repositioning Mode Menu and Cursor Hits</b>		
Cursor hit	Menu item	Function
<b>&lt;space&gt;</b>	At mouse	Reposition the point at the cursor cross hair position or closest grid point.
<b>C</b>	Give U, V, W	Switch to input from keyboard in cartesian coordinates UVW. Coordinates should be entered in free format. Default values of U, V and W are the values prior to the move. They can be accessed via the variables #1, #2 and #3. Type <b>Q</b> to leave point in its present position.
<b>E</b>	Remove C_line	Erase construction line closest to cursor cross hair.
<b>G</b>	Grid	Define or remove a grid (see page 4-55).
<b>H</b>		Display menu help message explaining all the cursor options.
<b>I</b>	Enter C_lines	Input construction line specifications.
<b>L</b>	List C_lines	List construction line specifications.
<b>N</b>	On nearest C_line	Reposition the point on the nearest construction line with minimum normal distance. This only affects the U and V coordinates; W remains unchanged.

<b>Point Repositioning Mode Menu and Cursor Hits (continued)</b>			
Cursor hit	Menu item	Function	
<b>P</b>	Give R, Theta, W	Switch to input from keyboard in local polar coordinates R $\theta$ W. Coordinates should be entered in free format. Default values of R, $\theta$ and W are the values prior to the move. They can be accessed via the variables #1, #2 and #3. Type Q to leave point in its present position.	
<b>Q</b>	Return without moving	Leave the point at its previous position.	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
		<b>&lt;return&gt;</b>	Reconstruct at the same size.
<b>T</b>	Show coordinates	Type the coordinates of the point closest to the cross-hairs, and select it for repositioning instead of the point already selected. The coordinates are given in cartesian (UVW) and polar (R $\theta$ W) coordinates.	

<b>Point Repositioning Mode Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>X</b>	At C_line intersection	Reposition the point at the closest construction line intersection or end. This only affects the U and V coordinates; W remains unchanged.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off (page 4-52).

### *Fast Polygon Input Menu and Cursor Hits*

In Fast Polygon Input Mode, points, lines and facets are all defined together. Points can be positioned at the cursor position and can be accurately positioned using construction lines or grids. Duplicate points are removed and new polygons can be defined using the points of existing polygons.

Mid-side points can be used to define polygon sides which are curved or have non-uniform element sizes. The mid-side points must be between the  $\frac{1}{4}$  and  $\frac{3}{4}$  points along the length of the side (this is verified by the **CHECK** command, page 4-18). If the mid-side point is not half way along the side the discretisation will also vary quadratically, with smaller elements near the corner which is closer to the mid-side point. This can be used to grade the mesh even for straight sided facets.

A mid-side point is defined after menu item **Next point is mid-side** or cursor hit **M** and can be defined using the cursor position, construction lines, grids or points of existing polygons. Mid-point definition can be cancelled using menu item **Cancel** or repeating cursor hit **M**. Points which have been used as corners cannot subsequently be used as mid-side points, and vice-versa.

Although the name *Polygon* is used, 4 sided facets can be defined for quadrilateral meshing.

<b>Fast Polygon Input Menu and Cursor Hits</b>		
Cursor hit	Menu item	Function
<b>&lt;space&gt;</b>	At mouse	Define a polygon corner at the cursor cross hair position or closest grid point.
<b>A</b>		Abort the <b>DEFINE</b> command.
<b>B</b>	Backtrack	Remove the last corner from the current polygon.

<b>Fast Polygon Input Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>D</b>	Delete facet	Delete the facet containing the cursor cross hair.
<b>E</b>	Remove C_line	Erase construction line closest to cursor cross hair.
<b>F</b>	Close polygon	Close an open polygon.
<b>G</b>	Grid	Define or remove a grid (see <a href="#">page 4-55</a> ).
<b>H</b>		Display menu help message explaining all the cursor options.
<b>I</b>	Enter C_lines	Input construction line specifications (see <a href="#">page 4-53</a> ).
<b>L</b>	List C_lines	List construction line specifications.
<b>M</b>	Next point is mid-side	The next point will be a mid-side point. It can be defined using <b>&lt;space&gt;</b> , <b>O</b> , <b>N</b> or <b>X</b> .
<b>N</b>	On nearest C_line	Define the next corner on the nearest construction line with minimum normal distance.
<b>O</b>	At nearest old point	Define next corner of polygon to be at the nearest point which has already been defined.
<b>Q</b>	Return	Return to Point Definition Mode.

Fast Polygon Input Menu and Cursor Hits ( <i>continued</i> )		
Cursor hit	Menu item	Function
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:
		4 numeric values <i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b> Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b> Return to previous screen size.
		<b>BOUND</b> Use bounding rectangle of geometry.
<b>&lt;return&gt;</b> Reconstruct at the same size.		

## Facet Definition Mode

Facets are defined by connecting the points on the base plane. Facets can be triangles, quadrilaterals or higher-order polygons with straight or curved edges. Points are selected in sequence around each facet:

- **corners**: menu items **Corner ... no auto-close** or **polygon corner** (cursor hits **C** or **P**). These are equivalent except that if a polygon corner is defined, the facet is not limited to four sides.
- **mid-side points**: menu item **Mid-side** (cursor hit **M**)

**Close** or cursor hit **F** can be used to close the facet. To extend a polygon to more than 4 sides, **polygon corner** or cursor hit **P** must be used for one of the first 4 corners.

Facets with 4 straight sides can be defined more quickly by selecting the 4 corner points with **Corner ... auto-close after 4** or cursor hit **<space>**.

Sides with mid-side points are quadratic. The mid-side points must be between the  $\frac{1}{4}$  and  $\frac{3}{4}$  points along the length of the side (this is verified by the **CHECK** command, page 4-18). If the mid-side point is not half way along the side the discretisation will also vary quadratically, with smaller elements near the corner which is closer to the mid-side point. This can be used to grade the mesh even for straight sided facets.

Points which have been used as corners cannot subsequently be used as mid-side points, and vice-versa.

Additional points can be defined by returning to Point Definition Modes (menu item **Return to points** or cursor hit **N**).

Facets can be copied and transformed using the Facet Group Operations Mode which can be entered using menu item **Go to Group Ops.** or cursor hit **G**.

Care must be taken to ensure that the entire base plane is covered with facets. This is especially so if the graphics facilities available do not shade facets as they are defined. When the base plane is complete, menu item **Go to Subdivisions** or cursor hit **Q** leaves Facet Definition Mode and moves on to Base Plane Subdivision Mode. It is possible to return to Facet Definition Mode from Base Plane Subdivision Mode in order to define more facets.

Full details of the cursor commands are given in the following section.

### *Facet Mode Menu and Cursor Hits*

Facet Mode Menu and Cursor Hits		
Cursor hit	Menu item	Function
<b>&lt;space&gt;</b>	auto-close after 4	Select nearest point as a corner. Facet closes automatically after 4 <b>&lt;space&gt;</b> hits.
<b>A</b>		Abort <b>DEFINE</b> command.
<b>C</b>	no auto-close	Select nearest point as a corner.
<b>D</b>	Delete facet	Delete the facet containing the cross-hairs.
<b>E</b>	Erase last point	Forget the last point selected for the current face.
<b>F</b>	Close	Close the current face (after at least 3 corners).
<b>G</b>	Go to Group Ops.	Enter Facet Group Operations Mode to group points for transformations or facets for copying.

<b>Facet Mode Menu and Cursor Hits (continued)</b>			
Cursor hit	Menu item	Function	
<b>H</b>		Display menu help message explaining all the cursor options.	
<b>M</b>	Mid-side	Select nearest point as a mid-side point. This cannot be used for the first point of a facet. The facet is closed automatically if the mid-side point is on the fourth side unless <b>P</b> has been used to define a corner.	
<b>N</b>	Return to points	Enter Point Definition Mode to define or move points.	
<b>P</b>	polygon corner	Select nearest point as a corner. Using this, rather than <b>&lt;space&gt;</b> or <b>C</b> identifies the facet as a polygon which can have more than 4 sides.	
<b>Q</b>	Go to Subdivisions	Leave the facet definition mode and move on to Base Plane Subdivision Mode.	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
<b>&lt;return&gt;</b>	Reconstruct at the same size.		

<b>Facet Mode Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off (page 4-52).

## Facet Group Operations Mode

There are two Group Operations which can be performed on the base plane: the coordinates of groups of points can be transformed and groups of facets can be copied:

- A point is added into the point group using menu item **Select/de-select point** or cursor hit **N**. This selects the nearest point or if it is repeated for a selected point it removes the point from the group.
- The coordinates of points in the point group can be transformed using menu item **Transform points** or cursor hit **T**. If there is no point group then all the points are transformed.
- A facet is added to the facet group using menu item **Select/de-select** or cursor hit **F** cursor hit. This selects the facet containing the cross-hairs or if it is repeated for a selected facet it removes the facet from the group.
- The facets in the facet group can be copied using menu item **Copy facets** or cursor hit **C**. If there is no facet group then all the facets are copied. Multiple copies can be made. The program prompts the user for the number of new copies, e.g. to end up with 8-fold symmetry it is necessary to create 7 new copies.
- The copy operation creates new facets, lines and points, and then transforms the coordinates of the points. For multiple copies the transformation applies between the original points and the first copy, between the first and second copies, between the second and third, etc. The index number of the copy (**#COPY**) can be used in transformation options **CARTESIAN** and **POLAR**.

At any one time there can be either a point group **or** a facet group, not both.

The transformation options are: **CARTESIAN**, **DISPLACE**, **MANGLE**, **MIRROR**, **POLAR**, **PROJECT**, **ROTATE** and **SCALE**.

When all transformations and copies are complete, menu item **Return to facets** or cursor hit **Q** is used to return to Facet Definition Mode. The program prompts

for a tolerance which is used to coalesce coincident points. Points are coalesced if the differences in U, V and W coordinates are all less than the tolerance.

Full details of the menu items, cursor commands and the transformation options are given in the following sections.

### *Group Operation Menu and Cursor Hits*

Group Operations Menu and Cursor Hits			
Cursor hit	Menu item	Function	
<b>C</b>	Copy facets	Copy the facet group, or all the facets if there is no group.	
<b>F</b>	Select/de-select facet	Select or de-select the facet containing the cross-hairs as a member of the facet group.	
<b>H</b>		Display menu help message explaining all the cursor options.	
<b>N</b>	Select/de-select point	Select or de-select the nearest point as a member of the point group.	
<b>Q</b>	Return to Facets	Leave the Facet Group Operations Mode and return to Facet Definition Mode.	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
		<b>&lt;return&gt;</b>	Reconstruct at the same size.

<b>Group Operations Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>T</b>	Transform points	Transform the point group, or all the points if there is no group.
<b>U</b>	Undo transform	Undo a point transformation operation.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off ( <a href="#">page 4-52</a> )

### Group Transformation Options

The transformations can be any combination of the following:

<b>Base Plane Transformation Commands</b>	
Command	Parameters and Function
<b>CARTESIAN</b>	<i>exp_u exp_v exp_w</i> The points are moved to new positions defined by expressions for their ( <i>u</i> , <i>v</i> and <i>w</i> ) coordinates in terms of their old coordinates: <b>U</b> , <b>V</b> , <b>R</b> , <b>TH</b> and <b>W</b> . The copy number, ( <b>#COPY</b> ), can also be used.
<b>DISPLACE</b>	<i>du dv dw</i> Displace points in the current viewing local coordinate system by adding ( <i>du</i> , <i>dv</i> , <i>dw</i> ) to their coordinates ( <i>u</i> , <i>v</i> , <i>w</i> ).
<b>MANGLE</b>	<i>u1 v1 angle</i> Reflect point coordinates in line specified by one point ( <i>u1</i> , <i>v1</i> ) and an <i>angle</i> . The <i>W</i> coordinate of the points is not affected.
<b>MIRROR</b>	<i>u1 v1 u2 v2</i> Reflect point coordinates in line specified by its end points ( <i>u1</i> , <i>v1</i> ), ( <i>u2</i> , <i>v2</i> ). The <i>W</i> coordinate of the points is not affected.
<b>POLAR</b>	<i>exp_r exp_θ exp_w</i> The points are moved to new positions defined by expressions for their ( <i>r</i> , <i>θ</i> and <i>w</i> ) coordinates in terms of their old coordinates: <b>U</b> , <b>V</b> , <b>R</b> , <b>TH</b> and <b>W</b> . The copy number, ( <b>#COPY</b> ), can also be used.
<b>PROJECT</b>	<i>du dv dw ucentre vcentre wcentre uangle vangle wangle</i> Project the points in the direction ( <i>du</i> , <i>dv</i> , <i>dw</i> ) until they intersect the XY plane of a coordinate system specified by its origin ( <i>ucentre</i> , <i>vcentre</i> , <i>wcentre</i> ) and axis rotation angles ( <i>uangle</i> , <i>vangle</i> , <i>wangle</i> ).
<b>QUIT</b>	End the sequence of transformations.

Base Plane Transformation Commands ( <i>continued</i> )	
Command	Parameters and Function
<b>ROTATE</b>	<i>ucentre vcentre wcentre uangle vangle wangle</i> Rotate points by angles ( <i>uangle</i> , <i>vangle</i> , <i>wangle</i> ) around axes parallel to the local coordinate system and passing through the point ( <i>ucentre</i> , <i>vcentre</i> , <i>wcentre</i> ).
<b>SCALE</b>	<i>ucentre vcentre factor</i> Scale points by multiplying the distance from local coordinate point ( <i>ucentre</i> , <i>vcentre</i> ) by <i>factor</i> . The W coordinate of the points is not affected.

- Example: to rotate about the origin by angle 30 degrees about the W axis:

```
OP-TRANS > r 0 0 0 0 0 0 30
```

- Example: to scale the model to half its size:

```
OP-TRANS > s 0 0 1/2
```

- Example: to move all points onto a circle, radius 10, leaving the azimuthal coordinates the same:

```
OP-TRANS > polar 10 atan2(v;u)*180/pi w
```

## Base Plane Subdivision Mode

This mode is omitted if **DEFINE** is being used for conductor elements.

The volumes created by extruding the initial surface plane (base plane) are treated as super-elements. They are normally divided into smaller elements that are used for the actual finite element approximation. The division of the volume into elements is determined by the subdivision defined for the facet edges and the subdivision specified for each layer of mesh created by an extrusion operation.

A default subdivision of 1 is initially set for all facet edges. When all the edges have been updated, the program will reply that the subdivision is complete.

- **If there are no polygonal facets**, regular subdivision is assumed. In this case, when the subdivision is given for a single edge and that edge is one side of a quadrilateral facet, then the subdivision of the opposite edge will be set to the same value. The subdivision will be carried through the mesh until a triangular facet, the edge of the mesh or a facet already set in this operation is reached.
- In models with regular subdivision, triangular facets are meshed by mapping onto a quadrilateral with 2 corners coincident. Therefore triangles must have

2 sides with the same number of subdivisions. The program can check whether this rule has been satisfied using menu item **Check for hex meshing** or cursor hit **C**. The check is also made if Subdivision Mode is left using cursor hit **Q**.

- **If there are polygonal facets**, the subdivision of each edge must be set individually unless all edges are set to the same subdivision.

The subdivision of each edge is uniform, unless the edge is a quadratic line with the mid-point not at the geometric mid-point of the line. In that case the elements at the end of the edge closer to the mid-point will be smaller than those at the other end.

- *In keyboard mode* a facet is selected for subdivision by positioning the cross-hairs just inside a facet, close to the edge to be divided and pressing the **<SPACE>** bar. The program prompts for the number of subdivisions. Subdivisions can also be set globally, i.e. all edges set to the same subdivision number, using the **G** cursor hit.
- *In menu mode* the subdivision can be set first using menu item **Set subdivision** and can then be applied to a single edge or globally to the entire mesh.

Subdivisions of edges can be changed any number of times, until the required pattern is achieved.

Additional facets can be defined by returning to Facet Definition Mode with menu item **Return to Facets** or menu hit **F**. If this facility is used it is essential to ensure that the mesh subdivision is complete before finally leaving Facet Subdivision Mode. Menu item **Go to Extrusions** or cursor hit **Q** leaves Facet Subdivision Mode and proceeds to Extrusion Mode. It is necessary to make sure that the extrusion direction required has been set using menu item **Change view** or cursor hit **V** before leaving the Subdivision Mode.

Full details of the cursor commands are given in the following section.

### *Facet Subdivision Menu and Cursor Hits*

<b>Facet Subdivision Menu and Cursor Hits</b>		
Cursor hit	Menu item	Function
<b>&lt;space&gt;</b>	Apply to line	Select closest edge for its subdivision to be set.
<b>A</b>		Abort the <b>DEFINE</b> command.
<b>C</b>	Check for hex meshing	If there are no polygons, check that the subdivisions obey the rules for meshing with quadrilaterals/hexahedra.

<b>Facet Subdivision Menu and Cursor Hits (continued)</b>			
Cursor hit	Menu item	Function	
<b>F</b>	Return to Facets	Return to Facet Definition Mode to define additional facets.	
<b>G</b>	Apply globally	Select all edges for their subdivisions to be set to the same value.	
<b>H</b>		Display menu help message explaining all the cursor options.	
<b>Q</b>	Go to Extrusions	Leave the Facet Subdivision Mode and proceed to the Extrusions Mode. If there are no polygons, the subdivisions are checked for quadrilateral/hexahedral meshing.	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
		<b>&lt;return&gt;</b>	Reconstruct at the same size.
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.	
<b>X</b>		Leave the Facet Subdivision Mode and proceed to the Extrusions Mode. No checking is done.	
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off ( <a href="#">page 4-52</a> ).	

## Extrusions Mode

Once the initial surface set of facets has been defined and subdivisions assigned to the edges, the program moves on to applying extrusion operations to the set of facets. There must be at least one extrusion operation applied to the set of facets, but many others may be needed to define the complete model. It is also possible to add more extrusions to a completed mesh using the **EXTEND** command. The **EXTEND** command must be used to add the second and subsequent extrusions if the data is defined in menu mode (page 4-113).

- *In keyboard mode* a single line command defines each extrusion. It consists of two compulsory numeric values with optional keywords. The first numeric value is interpreted as the W-coordinate for the points at the top of the new extrusion layer. The second numeric value gives the number of subdivisions between the planes. The keywords **LINEAR**, **QUADRATIC**, **GLOBAL** and **RELATIVE** specify the type of extrusion. The keyword **NOEDIT** indicates that no editing is required for points on the new mesh plane(s). The keyword **SUBDIVISION** request the program to return to Facet Subdivision Mode and ignore all other items on the command line. **SUBDIVISION** is only available before the first extrusion has been created.
- *In menu mode* a **LINEAR** or **QUADRATIC** extrusion can be chosen from a menu. The DialogBox shown in the section “**Extrusion DialogBox**” on page 4-73 allows the user to give the coordinate of the next plane, the number of subdivisions and to select **GLOBAL** or **RELATIVE**.

The values and options are interpreted as follows:

- **Coordinate of next plane.** The W-coordinate can be interpreted as **GLOBAL** or **RELATIVE**. When a **GLOBAL** coordinate is given for the new plane, the extrusion is formed by projecting the current facet set in the W direction until the W-directed lines intersect with the plane defined by the global value of W. The W-coordinate can also be interpreted as a **RELATIVE** move in the W direction, in which case the coordinate value is added to the W-coordinates of the points in the current plane.
- **Subdivision.** This specifies the number of layers of elements there will be between the two planes. This can be adjusted for the whole layer or for individual W-directed lines using the **MODIFY** command (page 4-135).  
The subdivision is ignored when **DEFINE** is being used for conductor elements.
- **Linear or Quadratic.** The W-directed lines can be **LINEAR** or **QUADRATIC**. The lines are created initially linear, i.e. straight, but if **QUADRATIC** is selected, a mid-extrusion plane of points is also created, at the geometric mid-points of the lines. This means that **QUADRATIC** lines can be changed to be curved, or to have non-uniform subdivision by moving the points on the

mid-extrusion plane away from the geometric mid-points of the lines. The mid-points should be between the  $\frac{1}{4}$  and  $\frac{3}{4}$  points along the length of the extrusion.

Unless **NOEDIT** has been requested, the coordinate values of points in the new plane, and in the mid-extrusion plane if **QUADRATIC** is selected, can be changed after the layer has been created. In some cases it is better to make the new plane(s) exactly the same as the current plane by using a **RELATIVE** coordinate of zero, and subsequently modifying the coordinates to the points using the Point Movement cursor options which are presented next. In other cases, it can be better to request **NOEDIT** and subsequently use the **MODIFY** (page 4-135) command to move the points.

For editing the point coordinates, the program draws a picture of the new plane and invites points to be moved individually or in groups. Individual points are selected using menu item **Move point** or cursor hit **<space>** and can be repositioned using the same options as offered for the base plane. Points can be grouped using menu item **Select/de-select point** or cursor hit **K** and transformed using menu item **Transform points** or cursor hit **T**. If no points have been grouped all the points of the plane will be transformed. The transformations are similar to those available in Facet Group Operations Mode.

The original coordinates are stored for all points moved or transformed so that they can be put back using the menu item **Undo move or transform** or cursor hit **U**.

- *In keyboard mode*, after the extrusion has been created and all necessary points have been moved, cursor hit **Q** moves on to the next extrusion. The user is given the option of creating another extrusion or not (valid replies are **YES** or **NO**). After a **NO** response, the program moves on to Material Definition Mode, or, when **DEFINE** is being use for conductor elements, copies the data into the conductor database for modification with the **CONDUCTOR** sub-commands (page 4-24).
- *In menu mode*, after the extrusion has been created and all necessary points have been moved, menu item **Finish editing** moves on to Material Definition Mode, or, when **DEFINE** is being use for conductor elements, copies the data into the conductor database for modification with the **CONDUCTOR** sub-commands (page 4-24). Further extrusions can be defined using the **EXTEND** command (page 4-113).

Full details of the extrusion command line, the point movement cursor commands and the transformation options are given in the following sections.

### Extrusion Command Lines.

Extrusion Command Lines	
Two compulsory numeric values:	
<i>first_value</i>	W coordinate ( <b>GLOBAL</b> or <b>RELATIVE</b> ) of next plane.
<i>second_value</i>	Number of subdivisions in the layer.
Optional keywords:	
<b>LINEAR</b> (default) or <b>QUADRATIC</b>	W-directed lines <b>LINEAR</b> (no 'Mid-extrusion Plane'), or <b>QUADRATIC</b> (with 'Mid-extrusion plane').
<b>GLOBAL</b> (default) or <b>RELATIVE</b>	W coordinate is <b>GLOBAL</b> or <b>RELATIVE</b> to previous plane.
<b>NOEDIT</b>	No point movements are required on the new plane(s).
<b>SUBDIVISIONS</b>	Return to Facet Subdivision Mode (only available before the first extrusion).

- Example: to extrude to W=10 with 3 subdivisions and quadratic W-directed lines:

```
OP-EXTRUDE > 10 3 global quadratic
```

- Example: to extrude by adding 5 to the W coordinates of the current plane with 4 subdivisions and linear W-directed lines; no point movements are necessary on the new plane:

```
OP-EXTRUDE > relative 5 4 linear noedit
```

### Extrusion DialogBox

In the DialogBox, the values of coordinate and number of elements must be supplied. **Global** and **Relative** are options.

Extrusion definition

Coordinate

Global  Relative

Number of elements

*Point Selection  
Menu and  
Cursor Hits*

Point Selection Menu and Cursor Hits		
Cursor hit	Menu Item	Function
<space>	Move point	Select point nearest cursor to be moved. It can be repositioned using the Point Repositioning Mode cursor hits (page 4-75).
H		Display menu help message explaining all the cursor options. (More help available after point selection.)
K	Select/de-select point	Select point nearest the cursor to be transformed. Repeating <b>K</b> for a selected point de-selects it.
Q	Finish Editing	Leave this plane. If on a mid-extrusion plane, move on to the top plane of the new layer. Otherwise the program asks about the next extrusion ( <i>keyboard mode</i> ) or moves on to material definitions ( <i>menu mode</i> ).
R	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:
		4 numeric values <i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b> Select diagonally opposite corners of the display area with <space> cursor hits.
		<b>RESTORE</b> Return to previous screen size.
		<b>BOUND</b> Use bounding rectangle of geometry.
<return>	Reconstruct at the same size.	

<b>Point Selection Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu Item	Function
<b>T</b>	Transform points	Define general transformations for the points on the plane. (Use <b>K</b> to select points, or all points will be transformed.) The transformation options are given in section <a href="#">“Plane Transformation Options” on page 4-77</a>
<b>U</b>	Undo move or transform	Undo the last move or transform operation.
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off ( <a href="#">page 4-52</a> ).

***The Point  
Repositioning  
Mode Menu and  
Cursor Hits***

<b>Point Repositioning Mode Menu and Cursor Hits</b>		
Cursor hit	Menu item	Function
<b>&lt;space&gt;</b>	At mouse	Reposition the point at the cursor cross hair position.
<b>C</b>	Give U, V, W	Switch to input from keyboard in cartesian coordinates UVW. Coordinates should be entered in free format. Default values of U, V and W are the values prior to the move. They can be accessed via the variables <b>#1</b> , <b>#2</b> and <b>#3</b> . Type <b>Q</b> to leave point in its present position.
<b>E</b>	Remove C_line	Erase construction line closest to cursor cross hair.
<b>H</b>		Display menu help message explaining all the cursor options.
<b>I</b>	Enter C_lines	Input construction line specifications.
<b>L</b>	List C_lines	List construction line specifications.

<b>Point Repositioning Mode Menu and Cursor Hits (continued)</b>			
Cursor hit	Menu item	Function	
<b>N</b>	On nearest C_line	Reposition the point on the nearest construction line with minimum normal distance. This only affects the U and V coordinates; W remains unchanged.	
<b>P</b>	Give R, Theta, W	Switch to input from keyboard in local polar coordinates R $\vartheta$ W. Coordinates should be entered in free format. Default values of R, $\vartheta$ and W are the values prior to the move. They can be accessed via the variables #1, #2 and #3. Type Q to leave point in its present position.	
<b>Q</b>	Return without moving	Leave the point at its previous position.	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
	<b>&lt;return&gt;</b>	Reconstruct at the same size.	

<b>Point Repositioning Mode Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>T</b>	Show coordinates	Type the coordinates of the point closest to the cross-hairs, and select it for repositioning instead of the point already selected. The coordinates are given in cartesian (UVW) and polar (R $\theta$ W) coordinates.
<b>X</b>	At C_line intersection	Reposition the point at the closest construction line intersection or end. This only affects the U and V coordinates; W remains unchanged.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off (page 4-52).

### Plane Transformation Options

The transformations can be any combination of the following:

<b>Plane Transformation Commands</b>	
Command	Parameters and Function
<b>CARTESIAN</b>	<i>exp_u exp_v exp_w</i> The points are moved to new positions defined by expressions for their ( <i>u</i> , <i>v</i> and <i>w</i> ) coordinates in terms of their old coordinates: <b>U</b> , <b>V</b> , <b>R</b> , <b>TH</b> and <b>W</b> . The copy number, ( <b>#COPY</b> ), can also be used.
<b>DISPLACE</b>	<i>du dv dw</i> Displace points in the current viewing local coordinate system by adding ( <i>du</i> , <i>dv</i> , <i>dw</i> ) to their coordinates (U, V, W).
<b>POLAR</b>	<i>exp_r exp_θ exp_w</i> The points are moved to new positions defined by expressions for their ( <i>r</i> , $\theta$ and <i>w</i> ) coordinates in terms of their old coordinates: <b>U</b> , <b>V</b> , <b>R</b> , <b>TH</b> and <b>W</b> . The copy number, ( <b>#COPY</b> ), can also be used.
<b>PROJECT</b>	<i>du dv dw ucentre vcentre wcentre uangle vangle wangle</i> Project the points in the direction ( <b>du</b> , <b>dv</b> , <b>dw</b> ) until they intersect the XY plane of a coordinate system specified by its origin ( <i>ucentre</i> , <i>vcentre</i> , <i>wcentre</i> ) and axis rotation angles ( <i>uangle</i> , <i>vangle</i> , <i>wangle</i> ).
<b>QUIT</b>	End the sequence of transformations.

Plane Transformation Commands <i>(continued)</i>	
Command	Parameters and Function
<b>ROTATE</b>	<i>ucentre vcentre wcentre uangle vangle wangle</i> Rotate points by angles ( <i>uangle</i> , <i>vangle</i> , <i>wangle</i> ) around axes parallel to the local coordinate system and passing through the point ( <i>ucentre</i> , <i>vcentre</i> , <i>wcentre</i> ).
<b>SCALE</b>	<i>ucentre vcentre factor</i> Scale points by multiplying the distance from local coordinate point ( <i>ucentre</i> , <i>vcentre</i> ) by <i>factor</i> . The W coordinate of the points is not affected.

*In menu mode* the transformation commands are offered as menu items and the parameters are supplied using parameter boxes.

- Example: to shift the selected points sideways by 5 units in the U direction:

```
OP-TRANS > displace 5 0 0
```

- Example: to rotate the plane by -30 degrees around an line parallel to the V axis and passing through W=10:

```
OP-TRANS > rotate 0 0 10 0 -30 0
```

- Example: to double the U and halve the V coordinates of the points:

```
OP-TRANS > cartesian u*2 v/2 w
```

## Material Definition Mode

The extrusion operations create a discretisation of space. It is now necessary to set the material and mesh properties within each volume. To do this, the program presents the user with each layer in turn and the user sets values which over-ride the default or current settings within each volume.

- *In keyboard mode* volumes are selected using the cursor. Single volumes are selected using cursor hit **<space>**. Multiple volumes are selected using **K** in each volume except the last. The last volume should be selected using **<space>**. After a **<space>** cursor hit the program prompts for a one line material property definition. A summary of the volume properties can be displayed in each volume using **S**. This gives the name, potential code and element type. A full list of all the properties of an individual volume can be obtained using **L**. Cursor hit **Q** moves on to the next layer and **F** finishes definition in all layers. After material definition the program moves on to Boundary Condition Definition Mode.

- *In menu mode* the menu item **Select/de-select volume** allows volumes to be added or removed from a list. Menu item **Select and define** adds one last volume to the list and causes the program to display a DialogBox into which the material properties can be entered. Other menu items allow the properties to be summarized in all the volumes (**Show**) or to be **Listed** for an individual volume.

Material property definitions consist of two compulsory keywords and several options which add special properties or control setting of several volumes simultaneously.

The first compulsory keyword is the **material name**. This can be (almost) any character string of up to 8 characters, beginning with a letter. There are two pre-defined material names, **AIR** and **NULL**. **AIR** is for any non-conducting volume with a relative permeability and relative permittivity both equal to 1. Volumes with name **NULL** are omitted from the final mesh, enabling the creation of holes in the mesh to represent e.g. electrodes. (The second keyword must be omitted for material name **NULL**.) The definition of each material in terms of its permeability and, if necessary, conductivity is supplied by the **MATERIALS** command (see “[The MATERIALS Command](#)” on page 4-129), or during the commands which create the analysis data files.

The second compulsory keyword is the **potential type**. The following rules must be followed, depending on which analysis program will be used:

- **TOSCA (magnetostatics)**

**REDUCED** scalar potential *must* be used in a space where source currents are flowing. It is often easiest to make all the **AIR** use reduced scalar potential but a model should not have only **REDUCED** potential volumes.

**TOTAL** scalar potential *should* be used in non-**AIR** volumes. It is also possible to use reduced scalar potential in non-**AIR** volumes but this should only be used as a last resort, if it is not possible to specify a region where the total scalar potential would be single valued.

The total scalar potential becomes multi-valued if there exists a closed path, entirely in total scalar potential volumes, through which a net current flows. TOSCA breaks such a path by the use of automatic cuts, i.e. additional volumes which can have multiple values of potential on the surface. These volumes are given additional labels, **POTENTIAL\_CUT<sub>n</sub>**, so that they can be selected and displayed in the post processor.

If automatic cuts are disabled ([page 4-169](#)), it is necessary to use reduced scalar potential volumes or two different total scalar potential boundary conditions with one value inside the coil and another outside (see “[Boundary Condition Definition Mode](#)” on page 4-86). In this case the potential differ-

ence between the boundaries should exactly balance the current enclosed in the mesh (*S.I.* units):

$$\Delta\psi = - \int \mathbf{H} \cdot d\mathbf{l} = \mathbf{I} \quad (4.2)$$

**TOTAL** scalar potential can also be used in any **AIR** volumes which do not contain source currents. This reduces the number of nodes at which the coil field has to be calculated and also improves the accuracy of the total field in situations where the fields from the coils and the magnetic materials almost cancel (shielding problems).

To achieve good answers, it is important that coil fields can be represented well by the finite element mesh on the interface between the reduced and other potentials. This implies that the best choice of reduced potential region is a simple shaped region that contains the coils and which has a surface as far as possible from the coil surface. It is not always possible to achieve this, especially when for example the coil is wrapped on an iron surface since in this case the interface must correspond to the coil surface. In such cases the discretisation of the interface should be increased to maintain accuracy and the choice of adaptive integration is essential.

Any **VECTOR** potential volumes in a TOSCA model will be treated as **TOTAL** scalar potential.

- **ELEKTRA and CARMEN**

**REDUCED** scalar potential *must* be used in a space where source currents are flowing. It is often easiest to make all the **AIR** use reduced scalar potential but a model should not have only **REDUCED** potential volumes.

**TOTAL** scalar potential *should* be used in non-conducting non-**AIR** volumes. It is also possible to use reduced scalar potential but this should only be used as a last resort, if it is not possible to specify a region where the total scalar potential would be single valued.

**TOTAL** scalar potential can also be used in any **AIR** volumes which do not contain source currents. This reduces the number of nodes at which the coil field has to be calculated and also improves the accuracy of the total field in situations where the fields from the coils and the magnetic materials almost cancel.

Magnetic **VECTOR** potential *must* be used in non-**AIR** volumes with non-zero conductivity. In some situations, it must also be used in adjacent **AIR** or other non-conducting volumes so that the total magnetic scalar potential in surrounding volumes is single valued.

In some models it is convenient to assign a current density to volumes of the mesh. These non-**AIR** volumes should be modelled with magnetic **VECTOR** potential and should have zero conductivity. The current density vector is described below.

The total magnetic scalar potential becomes multi-valued if there exists a closed path, entirely in total scalar potential volumes, through which a net current flows. In this context, an interface between vector potential and reduced scalar potential is an infinitely thin total potential volume, which might also include a closed path through which a net current flows. All such paths must be broken by the use of vector or two different total scalar potential boundary conditions with one value inside the coil and another outside (see “[Boundary Condition Definition Mode](#)” on page 4-86). In this case the potential difference between the boundaries should exactly balance the current enclosed in the mesh (*S.I.* units):

$$\Delta\psi = - \int \mathbf{H} \cdot d\mathbf{l} = \mathbf{I} \quad (4.3)$$

To achieve good answers, it is important that coil fields can be represented well by the finite element mesh on the interface between the reduced and other potentials. This implies that the best choice of reduced potential region is a simple shaped region that contains the coils and which has a surface as far as possible from the coil surface. It is not always possible to achieve this, especially when for example the coil is wrapped on an iron surface since in this case the interface must correspond to the coil surface. In such cases the discretisation of the interface should be increased to maintain accuracy and the choice of adaptive integration is essential.

- **SCALA and TOSCA (current flow and electrostatics)**

All volumes are treated as if they use **TOTAL** scalar potential.

- **SOPRANO**

All volumes must use magnetic **VECTOR** potential. The program overrides the assignment of other potential types.

More information on the use of the different potentials is given in the user guides for the analysis programs.

Optional keywords defining properties are element types (**LINEAR**, **QUADRATIC**), one scalar property (**SCALAR**), three vector properties (**VECTOR**, **CURRENT**, **VELOCITY**) and a lamination **PACKING** factor. The vector properties can be defined as expressions in terms of the coordinates (**X**, **Y** and **Z**) to allow spatial variation.

- **LINEAR** and **QUADRATIC** define the **element types** to be used. **LINEAR** elements are 8-noded hexahedra or 4 noded tetrahedra; **QUADRATIC** elements are 20-noded isoparametric hexahedra or 10-noded isoparametric tetrahedra. Both element types can be used in one model. (Only linear elements can be used in SOPRANO-EV.)
- In **electrostatic problems** (SCALA and TOSCA), the **volume electric charge density** is specified by **SCALAR** followed by one numeric value.

- In problems with **permanent magnets** the coercive force is specified by the BH curve, but the direction is specified here by **VECTOR** followed by 3 Euler angles (page 2-31) which define a local coordinate system for the volume. The permanent magnetic field is in the local negative Z direction.
- In problems with **anisotropic materials** the local coordinate system for the material is defined by **VECTOR** followed by 3 expressions representing the Euler angles (page 2-31). If the material is laminated, (TOSCA only) the **packing factor** is specified by **PACK** followed by one numeric value which modifies the permeability in the directions normal and parallel to the laminations (the local XY plane). If the material is to be specified by **multiple properties**, only the **VECTOR** property is required here to define the local coordinate system for the material. In either case, anisotropy must be switched on in the **MATERIAL** sub-command of **SOLVER** (page 4-153).
- In **eddy current problems** (ELEKTRA), an assigned current density can be specified by **CURRENT** followed by 3 expressions representing the x, y and z-components of the current density. This only applies to non-conducting **VECTOR** potential volumes.
- In **motion induced eddy current problems** (ELEKTRA-VL), the linear **velocity** is specified by **VELOCITY** followed by 3 expression representing the x, y and z-components of velocity in units of length sec<sup>-1</sup>. Rotational motion is given by **SCALAR** followed by the angular velocity in rpm. The rotation is always around the global Z axis.

The three values associated with **CURRENT**, **VECTOR** and **VELOCITY** can be given as expressions in terms of X, Y and Z so that vectors and properties which vary throughout the volumes can be specified. The computed directions can be viewed using the **DISPLAY** command (page 4-97).

Optional keywords for setting additional volumes with the same definition are **ALL**, **FROM**, **TO** and **KEEP**. **ALL** indicates that all the volumes in the layer(s) should be set the same. Both **FROM** and **TO** should be followed by numeric values specifying a layer number to specify a range of layers to be set the same. If either is omitted, the current layer number is assumed. The numeric value after **TO** can be specified as \* to indicate the top layer. **KEEP** specifies that the current selection of volumes should be kept, following the definition of materials, so that another definition can be given. This can be used when setting corresponding volumes in a non-contiguous set of layers.

The material names, potential types and element types are stored as labels on each volume and can be used to select parts of the model in the **DISPLAY** command (page 4-97).

Full details of the cursor commands and material definitions are given in the following sections.

*Material  
Definition Mode  
Menu and  
Cursor Hits*

<b>Material Definition Menu and Cursor Hits</b>		
Cursor hit	Menu item	Function
<b>&lt;space&gt;</b>	Select and define	Select the volume containing the cross-hairs to give a material definition for this volume and any selected with <b>K</b> .
<b>F</b>	Finish	End the volume material definition mode and move on to Boundary Condition Mode.
<b>H</b>		Display menu help message explaining all the cursor options. (More help available after volume selection.)
<b>K</b>	Select/de-select volume	Keep the volume containing the cross-hairs until a <b>&lt;space&gt;</b> is used. Repeating <b>K</b> for a selected volume, de-selects the volume.
<b>L</b>	List properties	List all the properties of the volume containing the cross-hairs.
<b>Q</b>		Move on to the next extrusion layer or, if at the last layer move on to Boundary Condition Mode.

<b>Material Definition Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:
		4 numeric values <i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b> Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b> Return to previous screen size.
		<b>BOUND</b> Use bounding rectangle of geometry.
<b>&lt;return&gt;</b> Reconstruct at the same size.		
<b>S</b>	Show volumes	Display the material names, potential codes and element types.
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off (page 4-52).

**Material Definition Command Line**

The single line definitions of material and other volume properties consist of the following items:

<b>Material Definitions</b>	
Two compulsory keywords:	
<i>first_keyword</i> or <b>QUIT</b>	Material name. Use <b>AIR</b> for air, <b>NULL</b> to omit volume from mesh or a material name. <b>QUIT</b> abandons the currently selected volumes.

<b>Material Definitions (continued)</b>	
<i>second_keyword</i>	Potential type: <b>REDUCED</b> , <b>VECTOR</b> or <b>TOTAL</b> . It need not be given for material <b>NULL</b> .
Optional keywords – element types:	
<b>LINEAR</b> (default) or <b>QUADRATIC</b>	<b>LINEAR</b> (8-noded) elements, or <b>QUADRATIC</b> (20-noded) elements.
Optional keywords – volume properties:	
<b>PACK value</b>	Sets packing factor for volume.
<b>SCALAR value</b>	Sets charge density (electrostatics only). Sets angular velocity in <b>VECTOR</b> potential volumes [rpm].
<b>CURRENT jx jy jz</b>	Sets assigned current density in <b>VECTOR</b> potential volumes.
<b>VECTOR t p s</b>	Sets local coordinate system Euler angles of volume for permanent magnets or anisotropic materials.
<b>VELOCITY vx vy vz</b>	Sets linear velocity in <b>VECTOR</b> potential volumes [length unit sec <sup>-1</sup> ].
Optional keywords – setting multiple volumes:	
<b>ALL</b>	Sets all volumes in layer(s).
<b>FROM value</b>	Sets all layers from layer number given though to the current layer or the layer number given with <b>TO</b> .
<b>KEEP</b>	Keeps current selection of volumes for another material definition.
<b>TO value</b>	Sets all layers from current layer or the layer number given with <b>FROM</b> though to the layer number given. The value may be * to indicate the top layer.

- Example: to set selected volumes to material name **IRON** in layers 3, 4,..., 7 and 8:

```
OP-MATERIALS > iron total from 3 to 8
```

- Example: to set all volumes in the layer to material name **ALCOMAX** with vector direction set:

```
OP-MATERIALS > alcomax total all vect 90 90 0
```

### Material Definition DialogBox

In the DialogBox, the default properties correspond to the last volume selected. A material name must be supplied. Unless the material is **NULL**, one of the potential types must be selected. All the other items are optional and should only be assigned values if needed. The **Local XYZ** options allow the **VECTOR** property to be set to appropriate Euler angles to swap the coordinate axes. If another direction is needed, the Euler angles should be typed into the **Other vector** box.

Material Definition

Material Name

Potential Type:  Total Scalar  Reduced Scalar  Vector

Element Type:  Linear  Quadratic

Options:

Jx, Jy, Jz

Vx, Vy, Vz

Scalar: Charge Density or Rotational Velocity

Scalar

Packing factor

Material orientation

Local XYZ=XYZ  Local XYZ=YZX  Local XYZ=ZXY

Other vector

Other volumes and layers:

From  To   All volumes

## Boundary Condition Definition Mode

Boundary conditions can be set on any surface of any volume in the mesh. In order to achieve this the program presents the facets in separate groups:

*In keyboard mode* there are 4 groups of facets: first the facets on the base plane; second the planes between the first and the last; third the final plane of the mesh; and fourth the extrusion facets normal to the planes, one layer at a time. Facets are selected using the cursor. Single facets are selected using **<space>**. Multiple facets are selected using **K** for each facet except the last. The last facet should be

selected using **<space>**. After a **<space>** cursor hit the program prompts for a one line boundary condition definition. Alternatively all the free surfaces of the mesh (all planes and layers) can be set to the same boundary condition using **G**. Free surfaces are those which are on the outside of the mesh or adjacent to a volume with material name **NULL**. A full list of all the conditions on an individual facet can be obtained using **L**. Cursor hit **Q** moves on to the next plane or layer and **S** or **F** finishes definition in all planes or layers. After boundary condition definitions the **DEFINE** command is complete and the program expects another top-level command.

*In menu mode* there are 3 groups of facets: first the facets on the base plane; second the top plane of the mesh; and third the extrusion facets normal to the planes. The menu item **Select/de-select facet** allows facets to added or removed from a list. Menu item **Select and define** adds one last facet to the list and causes the program to display a DialogBox into which the boundary conditions can be entered. The menu item **List conditions** allows the boundary conditions to be listed for an individual facet.

Boundary conditions can be imposed in seven ways:

- by restricting the **MAGNETIC** or **ELECTRIC** field to be **NORMAL** or **TANGENTIAL**. This sets combinations of the potentials and derivatives as appropriate for the type of problem and volume concerned:

<b>Normal and Tangential Boundary Conditions</b>			
<b>Normal Magnetic</b>	<b>Tangential Magnetic</b>	<b>Normal Electric</b>	<b>Tangential Electric</b>
<b>TOSCA Magnetic Field Problem</b>			
$\psi = 0$	$\frac{\partial \psi}{\partial n} = 0$	not allowed	not allowed
<b>SCALA or TOSCA Electric Field Problem</b>			
not allowed	not allowed	$V=0$	$\frac{\partial V}{\partial n} = 0$
<b>CARMEN/ELEKTRA Scalar potential Volumes</b>			
$\psi = 0$	$\frac{\partial \psi}{\partial n} = 0$	$\frac{\partial \psi}{\partial n} = 0$	$\psi = 0$
<b>CARMEN/ELEKTRA/SOPRANO Vector potential Volumes</b>			
$\mathbf{A} \cdot \mathbf{n} = 0$	$\mathbf{A} \times \mathbf{n} = 0, V = 0$	$\mathbf{A} \times \mathbf{n} = 0, V = 0$	$\mathbf{A} \cdot \mathbf{n} = 0$

In the above table  $\psi$  indicates the total or reduced magnetic scalar potential,  $\mathbf{A}$  the magnetic vector potential and  $V$  the voltage.

If necessary a potential boundary condition can be assigned in addition to the normal or tangential conditions in order to override the zero values of scalar potential.

- by setting values of magnetic scalar **POTENTIAL**, the **VOLTAGE** or components of the magnetic vector potential (**AX**, **AY** and **AZ**). For SOPRANO, the incident values of magnetic vector potential and voltage can be specified instead (**INAX**, **INAY**, **INAZ** and **INVO**).
- by setting the normal derivative of the magnetic scalar or vector potentials (**DERIVATIVE**, **DAX**, **DAY** and **DAZ**) or a mixed magnetic scalar potential condition in TOSCA (**PMIX**).
- by imposing a **RADIATION** condition (SOPRANO only).
- by imposing a **PEC** (perfect conductor) boundary condition (CARMEN, ELEKTRA and SOPRANO only). This is functionally equivalent to **NORMAL ELECTRIC** but allows boundaries to be distinguished in the post processor.
- by imposing a **SLIP** condition (CARMEN only) to identify the interface between the stator and rotor.
- by setting a **SYMMETRY** or periodicity condition (TOSCA and SCALA only).

If potential or derivative boundary conditions are used, a facet can have up to 4 potential boundary conditions – one for the scalar potential and one for each component of the vector potential. Thus, for example, defining a condition on the normal derivative of the X component of the vector potential would overwrite any previously defined condition on the  $A_x$ , but would not affect any conditions on  $A_y$ ,  $A_z$  or the scalar potential.

Boundary condition definitions consist of a compulsory keyword and up to two values followed by several options which control setting of several facets simultaneously or clearing boundary conditions.

The compulsory keyword is the boundary condition name. This can be one of **NORMAL**, **TANGENTIAL**, **POTENTIAL**, **VOLTAGE**, **AX**, **AY**, **AZ**, **INAX**, **INAY**, **INAZ**, **INVO**, **DERIVATIVE**, **DAX**, **DAY**, **DAZ**, **PMIX**, **RADIATION**, **PEC**, **SLIP** or **SYMMETRY**. The boundary condition name is assigned to the facet as a label. If normal or tangential boundary conditions are applied, the following labels are assigned to the facet as appropriate: **ELECTRIC**, **MAGNETIC**, **NORMELEC**, **NORMMAGN**, **TANGELEC**, **TANGMAGN**.

**CLEAR** followed by a boundary condition name clears that condition from the facet(s).

Boundary conditions have the following effects:

- In **magnetic scalar potential volumes**, setting a constant **POTENTIAL** condition specifies that the tangential components of the field are to be zero. Non-zero total scalar **POTENTIAL** conditions can be used to impose an external field (i.e. an m.m.f. across the model) or to balance enclosed currents in multiply connected geometries. Reduced scalar **POTENTIAL** boundary conditions can only have the value zero.

A constant **DERIVATIVE** specifies the value of the normal field (TOSCA only). The zero **DERIVATIVE** condition is the natural condition of the finite element mesh and need not be assigned explicitly.

- In **electric scalar potential volumes** (electrostatics or current flow), setting a constant **VOLTAGE** condition specifies that the tangential components of the field are to be zero. Non-zero total **VOLTAGE** conditions can be used to impose an external field (i.e. a potential difference across the model).

A constant **DERIVATIVE** specifies the value of the normal field. The zero **DERIVATIVE** condition is the natural condition of the finite element mesh and need not be assigned explicitly.

- In **vector potential volumes**, setting **AX**, **AY** or **AZ** to zero implies that the electric field in that direction is also zero. The constant electric **VOLTAGE** boundary condition should also be set on surfaces where the tangential components of electric field are zero. This constant should in general be zero, unless the voltage is providing the driving field.

A second set of boundary conditions (**INAX**, **INAY**, **INAZ**) set the characteristics of the incident field for SOPRANO and also apply the **RADIATION** condition

- **Periodic or SYMMETRY boundaries** (TOSCA and SCALA only) specify that the potentials on one surface are equal to the values on another surface (with or without a change of sign). The facets forming such surfaces are labelled here with the boundary condition **SYMMETRY** and are joined together in the **PERIODICITY** sub-command of the **SOLVERS** command (page 4-165) by the specification of transformations which map one set of boundary nodes onto the other.
- **SLIP boundaries** (CARMEN only) identify the interface between the stationary and rotating parts of a rotating machine. The volumes on both sides of a slip surface should be modelled using **TOTAL** scalar potential. It is often more convenient to apply the **SLIP** boundary condition using the **SLIP** command (page 4-152).

In some circumstances it may be necessary to specify the boundary condition values as functions of the node coordinates. The values of conditions **POTENTIAL**, **VOLTAGE**, **AX**, **AY** and **AZ** can be specified as expressions in terms of **X**, **Y** and **Z**. User variables and system constants (**PI**, **MU0**, **EPSILON0** and **C**) can be used in the expressions. The program assigns names for boundary condition expressions and these names can be used as labels in the **DISPLAY** command (page 4-97).

For analysis with CARMEN, ELEKTRA and SOPRANO/SS, non-zero potential boundary conditions (**POTENTIAL**, **VOLTAGE**, **AX**, **AY** or **AZ**) can be given a drive label so that the associated time-function or phase angle can be assigned by the **SOLVER** command (page 4-153).

Further information on boundary conditions is given in the User Guides for CARMEN, ELEKTRA, SCALA, SOPRANO and TOSCA and in the OPERA-3d Training Course.

The boundary condition names are stored as labels on each facet and can be used to select parts to the model in the **DISPLAY** command (page 4-97).

Optional keywords for setting additional facets with the same definition are **ALL**, **FROM** and **TO**. **ALL** indicates that all the facets in the plane(s) or layer(s) should be set the same. Both **FROM** and **TO** should be followed by a numeric value specifying a plane or layer number to specify a range of planes or layers to be set the same. If either is omitted, the current plane or layer number is assumed. The numeric value after **TO** can be specified as **\*** to indicate the highest numbered plane or layer.

Full details of the menu items and cursor commands and boundary condition definitions are given in the following sections.

**Boundary  
Condition  
Definition Mode  
Menu and  
Cursor Hits**

<b>Boundary Condition Mode Menu and Cursor Hits</b>		
Cursor hit	Menu item	Function
<b>&lt;space&gt;</b>	Select and define	Select the facet nearest the cross-hairs to give a boundary condition definition for this facet and any selected with <b>K</b> .
<b>F</b>		End the boundary condition definition mode (extrusion layers only).

<b>Boundary Condition Mode Menu and Cursor Hits (continued)</b>			
Cursor hit	Menu item	Function	
<b>H</b>		Display menu help message explaining all the cursor options. (More help available after facet selection.)	
<b>K</b>	Select/de-select facet	Keep the facet containing the cross-hairs until a <b>&lt;space&gt;</b> is used. Repeating <b>K</b> for a selected facet, de-selects the facet.	
<b>L</b>	List conditions	List all the boundary condition on the facet containing the cross-hairs.	
<b>Q</b>	Finish	Move on to the next plane or layer or, if at the last layer, leave Boundary Condition Mode.	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
	<b>&lt;return&gt;</b>	Reconstruct at the same size.	
<b>S</b>		Skip to the last plane (planes only).	

<b>Boundary Condition Mode Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off (page 4-52).

**Boundary  
Condition  
Command Line  
Definitions**

The single line commands for boundary condition definition consist of the following items:

<b>Boundary Condition Definitions</b>	
One compulsory keyword and values - one of the following:	
<b>AX</b> <i>expression label</i>	The X component of the vector potential.
<b>AY</b> <i>expression label</i>	The Y component of the vector potential.
<b>AZ</b> <i>expression label</i>	The Z component of the vector potential.
<b>CLEAR</b> <i>comp</i>	Clear boundary conditions with <i>comp</i> equal to <b>POTENTIAL</b> , <b>AX</b> , <b>AY</b> or <b>AZ</b> . If <i>comp</i> is omitted all boundary conditions are cleared.
<b>DAX</b> <i>value</i>	The normal derivative of $A_x$ (value must be zero).
<b>DAY</b> <i>value</i>	The normal derivative of $A_y$ (value must be zero).
<b>DAZ</b> <i>value</i>	The normal derivative of $A_z$ (value must be zero).
<b>DERIVATIVE</b> <i>value</i>	The normal derivative of scalar potential.
<b>INAX</b> <i>value label</i>	The incident vector potential, X component (SOPRANO).
<b>INAY</b> <i>value label</i>	The incident vector potential, Y component (SOPRANO).
<b>INAZ</b> <i>value label</i>	The incident vector potential, Z component (SOPRANO).
<b>NORMAL</b> <i>field</i>	Boundary conditions which restrict <i>field</i> to be normal to the facet. <i>Field</i> can be <b>MAGNETIC</b> or <b>ELECTRIC</b> .
<b>PEC</b>	Perfect conductor boundary condition (CARMEN, ELEKTRA and SOPRANO only).

<b>Boundary Condition Definitions (continued)</b>	
<b>PMIX</b> <i>value1 value2</i>	Mixed scalar potential boundary condition in TOSCA ( $\phi + \text{value1} \frac{\partial \phi}{\partial n} = \text{value2}$ )
<b>POTENTIAL</b> <i>expression label</i>	The magnetic scalar potential.
<b>QUIT</b>	Abandons the current selection of faces.
<b>RADIATION</b>	Radiation boundary condition (SOPRANO).
<b>SLIP</b>	Slip surface periodic boundary condition for a rotating machine.
<b>SYMMETRY</b>	Periodic boundary condition. Facets are connected by rotation and displacement operations defined in the <b>SOLVER</b> command (page 4-153).
<b>TANGENTIAL</b> <i>field</i>	Boundary conditions which restrict <i>field</i> to be tangential to the facet. <i>Field</i> can be <b>MAGNETIC</b> or <b>ELECTRIC</b> .
<b>VOLTAGE</b> <i>expression label</i>	The electric scalar potential.
Optional keywords - setting multiple facets:	
<b>ALL</b>	Sets all facets in plane(s) or layer(s).
<b>FROM</b> <i>value</i>	Sets all planes or layers from number given though to the current plane or layer or the number give with <b>TO</b> .
<b>KEEP</b>	Keeps current selection of facets for another boundary condition definition.
<b>TO</b> <i>value</i>	Sets all plane or layers from current plane or layer or the number given with <b>FROM</b> , through to the number given. The value may be * to indicate the top plane or layer.

- Example: to set selected facets to zero scalar potential:

OP-B/C > pote 0

### **Boundary Condition DialogBox**

The default settings correspond to the last facet selected.

The boundary condition can be specified in one of the following ways:

- Potentials: the option should be selected and a value given. For steady-state ac or transient analysis, a label can also be given to enable drive information to be associated with the boundary condition.

- Other boundary condition option buttons: no value is necessary.
- **CLEAR**: the condition to be cleared can be given in the value box.

Other types of boundary condition can be given by not selecting any of the options but giving a keyboard style boundary condition command line in the value box.

Boundary Conditions		
Condition name:		
<input type="checkbox"/> Magnetic Scalar	<input checked="" type="checkbox"/> Normal Magnetic	<input type="checkbox"/> Tangential Magnetic
<input type="checkbox"/> Voltage	<input type="checkbox"/> Normal Electric	<input type="checkbox"/> Tangential Electric
<input type="checkbox"/> Total Ax	<input type="checkbox"/> Total Ay	<input type="checkbox"/> Total Az
<input type="checkbox"/> Incident Ax	<input type="checkbox"/> Incident Ay	<input type="checkbox"/> Incident Az
<input type="checkbox"/> Incident Voltage	<input type="checkbox"/> Perfect Conductor	<input type="checkbox"/> Radiation
<input type="checkbox"/> Normal Derivative	<input type="checkbox"/> Mixed Derivative	
<input type="checkbox"/> Symmetry	<input type="checkbox"/> Slip Surface	<input type="checkbox"/> Clear
Value <input type="text"/>	Label/2nd value <input type="text"/>	
Other volumes and layers:		
From <input type="text"/>	To <input type="text"/>	<input checked="" type="checkbox"/> All facets
<input type="button" value="Accept"/>	<input type="button" value="Keep"/>	<input type="button" value="Help"/>
		<input type="button" value="Quit"/>

## The **DEVICE** Command

---

### Menu Route:

OPTIONS↓  
Graphics output

### Command Line Parameters:

Command	<b>DEVICE</b>
No Parameters	

There are two different graphics implementations of the software:

- **Windows** (available on Windows95, Windows98 and Windows NT4, Windows 2000 and Windows ME operating systems)
- **X-lib** (available on UNIX operating systems).

The **DEVICE** command is only available with X-lib graphics.

Four Graphics Options are available when the program is started and two of them can be reselected using the **DEVICE** command. The options are:

Option	Start-up	<b>DEVICE</b> command	Meaning
<b>SCREEN</b>	*	*	graphics displayed on the screen
<b>FILE</b>	*		all graphics commands stored in one file
<b>BOTH</b>	*	*	graphics on the screen and in a file
<b>NONE</b>	*		no graphics, except that the <b>DUMP</b> command can still be used to create picture files of specific pages.

If the program is started with no screen graphics (**FILE** or **NONE**), the GUI is not available and cannot be made available by the **DEVICE** command.

The initial specification of the graphics option can be stored in an environment variable (UNIX), in both cases called **VFGRAPHICS**. A valid value of **VFGRAPHICS** eliminates the initialization prompt.

Other environment variables which affect the software on UNIX are:

Variable	Meaning
VFWINDOWW	the initial window width in pixels
VFWINDOWH	the initial window height in pixels
VFINV	If this is set to <b>INVERT</b> , the initial setting of text and background colours will be black on white instead of the default of white on black.

On Windows systems, similar functionality can be obtained using the **Options** → **Graphics Window Preferences** menu item on the OPERA Console window.

Picture files can be read by the PICOUT program which is supplied with UNIX versions of the software and is described with the **DUMP** command.

Picture files can also be created using the **DUMP** command (“[The DUMP Command](#)” on page 4-104).

## The **DISPLAY** Command

---

### Menu Route:

DISPLAY↓  
3d Viewer ... refresh display

### Command Line Parameters:

Command	<b>DISPLAY</b>		
Parameter	Default	Function	
<b>SIZE</b>	10	Size of coordinate space to be displayed. The space extends from ( <b>XORIGIN</b> , <b>YORIGIN</b> , <b>ZORIGIN</b> ) by <b>SIZE</b> in each direction.	
<b>XEYE</b>	0	X-coordinate of eye position.	
<b>YEYE</b>	0	Y-coordinate of eye position.	
<b>ZEYE</b>	100	Z-coordinate of eye position.	
<b>PERSPECTIVE</b>	<b>NO</b>	Perspective view switch.	
		<b>NO</b>	Orthogonal projection.
		<b>YES</b>	Perspective projection.
<b>ROTX</b>	0	Rotation angle about X-axis to change eye position.	
<b>ROTY</b>	0	Rotation angle about Y-axis to change eye position.	
<b>ROTATE</b>	0	Rotation angle about viewing direction.	
<b>ELEMENT</b>	<b>NO</b>	Element display switch.	
		<b>NO</b>	No element subdivision.
		<b>SURFACE</b>	Subdivision on volume surfaces.
		<b>VOLUME</b>	Subdivision within volumes.
<b>MESH</b>	<b>ALL</b>	Mesh number, <b>ALL</b> or <b>NONE</b> .	
<b>TYPE</b>	<b>VOLUME</b>	Type(s) of entities to be displayed. <b>VOLUME</b> , <b>FACET</b> , <b>LINE</b> , <b>POINT</b> , <b>ALL</b> or <b>SAME</b> .	

Command	DISPLAY (continued)		
Parameter	Default	Function	
LABEL	NOTAIR	Label(s) on entities.	
L1	1	First layer to be displayed.	
L2	*	Last layer to be displayed. * means top layer.	
COILS	YES	Conductor display switch.	
		NO	Conductors not displayed.
		YES	Conductors displayed.
C1	1	First conductor to be displayed.	
C2	*	Last conductor to be displayed. * means highest numbered conductor.	
XORIGIN	0	X-coordinate at centre of display.	
YORIGIN	0	Y-coordinate at centre of display.	
ZORIGIN	0	Z-coordinate at centre of display.	
HIDDEN	NO	Hidden surfaces removed switch.	
		FULL	Colour-fill display of visible surfaces (slower but more reliable algorithm).
		NO	Wire frame display of all surfaces.
		YES	Colour-fill display of visible surfaces

Command	<b>DISPLAY</b> (continued)	
Parameter	Default	Function
<b>VECTORS</b>	<b>CONDUCTORS</b>	Vector display switch.
		<b>CONDUCTORS</b> Vectors show current direction on conductors.
		<b>CURRENT</b> Vectors show current density direction.
		<b>MATERIAL</b> Vectors show material orientation.
		<b>NO</b> Vectors not displayed.
	<b>VELOCITY</b> Vectors show velocity direction.	
<b>ERASE</b>	<b>YES</b>	Picture erase switch.
		<b>NO</b> Old picture not erased.
	<b>YES</b> Old picture erased.	
<b>THREED</b>	<b>NO</b>	Copy view from 3d Viewer.
		<b>NO</b> Use <b>DISPLAY</b> command parameters.
	<b>YES</b> Set <b>DISPLAY</b> command parameters to match 3d Viewer.	
<b>AXES</b>	<b>YES</b>	Draw axes switch:
		<b>NO</b> No axes drawn.
	<b>YES</b> Display coordinate axes.	

The **DISPLAY** command draws pictures of the three dimensional geometry of the finite element mesh and conductors. Pictures can be line-drawings or coloured surfaces with hidden surfaces obscured. The discretisation can be included and vectors can be drawn to indicate the vector properties of the volumes and the current directions in the conductors. Colours are used to differentiate between volumes with different material names or to show values of assigned potentials. The parameters control the coordinate limits of the volume of three dimensional space

included, the direction of the view, the parts of the model included and other options.

Pictures can also be displayed using the 3d Viewer. (See “The **THREED** Command” on page 4-171.)

## The Coordinate Limits and the Viewing Direction

### *Menu Route:*

```
DISPLAY↓
  Display Command ... view
  Display Command ... copy 3d view
```

The **DISPLAY** command draws a picture including parts of the model which are inside a volume which is a cube of dimension **2\*SIZE**. The centre of the cube is at coordinates (**XORIGIN**, **YORIGIN**, **ZORIGIN**).

The view obtained is controlled by the eye position (**XEYE**, **YEYE**, **ZEYE**), the origin (**XORIGIN**, **YORIGIN**, **ZORIGIN**) and whether perspective is selected (**+PERSPECTIVE** or **-PERSPECTIVE**). For non-perspective (orthogonal) views the view direction is set by **XEYE**, **YEYE** and **ZEYE**. For perspective views the view direction is set by the eye position and the origin. The distance between the origin and the eye position is important; parts of the model behind the eye will be omitted from the display.

The viewing direction can also be altered by the rotation angle parameters (**ROTX** and **ROTY**), which have the effect of altering the values of **XEYE**, **YEYE** and **ZEYE**, the eye position. The values of **ROTX** and **ROTY** are not remembered. **ROTATE** specifies a rotation angle about the viewing direction.

### *Copying 3d View*

The above parameters (**SIZE**, **XEYE**, **YEYE**, **ZEYE**, **PERSPECTIVE**, **ROTX**, **ROTY**, **ROTATE**, **XORIGIN**, **YORIGIN** and **ZORIGIN**) can also be set so that the view of the model corresponds to that of the 3d Viewer (**+THREED**). The **THREED** parameter is set back to **NO** after each time it is used so that these parameters can be adjusted for the next **DISPLAY**.

The view parameters are only effective if the previous picture is erased with **+ERASE**. If the previous picture is not erased (**-ERASE**) then the view remains as before.

The coordinate axes can be optionally displayed to show the scale of the model (**+AXES**)

## Selecting Parts of the Finite Element Model

### Menu Route:

DISPLAY↓  
 Display Command ... select parts

The **DISPLAY** command draws all entities, volumes, facets, lines and points, created with the **DEFINE** command (page 4-97). By use of the **TYPE** and **LABEL** parameters, sub-sets of the entities can be selected in order to aid visualization of the model. The parameter **TYPE** can be set to any individual entity type or any combination of entity types separated by + or -. For example, to display lines and points, **TYPE=LINE+POINT** should be used; for everything except points, **TYPE=ALL-POINT**. Similarly, **LABEL** can be set to individual labels or combinations of labels. The additional label, **NOTAIR** can be used to select all material name labels except **AIR**. Abbreviated label names can be used where the abbreviation is not ambiguous. Abbreviations can be followed by \* to indicate that all labels which match should be selected. An entity is drawn if it has any of the labels selected. For example, to display all facets with vector potential boundary conditions, **LABEL=A\*-ALL** could be used.

Labels are assigned automatically to parts of the model by the commands **DEFINE**, **EXTEND**, **MODIFY** and **SLIP**. Automatically assigned labels include **ALL**, material names, element types potential types and boundary condition types and functions. Additional labels can be given in two ways: the **LABEL** command (page 4-125) can be used to give any label to any part of the model and the **CHECK** command (page 4-18) can be used to assign the labels **DEBUG** to volumes with bad shapes and **EXTERNAL** to facets which are not shared by two volumes.

At the end of the **DISPLAY** command, **LABEL** is set to **SAME** indicating that the same labels will be used for the next picture, unless **LABEL** is reset. **SAME** can be used in further combinations of labels to add or remove labels from the previously selected list.

The part of the finite element mesh displayed can also be restricted by the parameters **MESH**, **L1** and **L2**. The **MESH** parameter is used to select **ALL** meshes or one particular mesh (each **DEFINE** command creates a mesh with a new number). **MESH=NONE** can be used to omit the finite element mesh completely. **L1** and **L2** are used to select a subset of the layers of the mesh. **L2** can be set to \* to indicate the top layer.

## Selecting Conductors

### Menu Route:

```
DISPLAY↓
  Display command ... select parts → Conductors
  Display command ... select parts → Conductor numbers
```

The **DISPLAY** command draws the conductors including any symmetry copies. This can be controlled by the parameters **COIL**, **C1** and **C2**. **+COIL** and **-COIL** switch the display of the conductors on and off, and **C1** and **C2** select a range of conductors for display. **C2** can be set to **\*** to indicate the highest numbered conductor.

## Other parameters

### Menu Route:

```
DISPLAY↓
  Display ... command style
```

Two types of picture can be produced: wire-frame line drawings or coloured surface pictures, with hidden surfaces obscured. **-HIDDEN** produces a wire-frame picture; **+HIDDEN** uses a fast but unsophisticated algorithm to produce the “hidden surface” pictures. This does not always achieve a perfect picture. It is usually possible to obtain a satisfactory picture with a suitable choice of view point or **HIDD=FULL** which orders the displayed facets by a slower but more reliable algorithm could be used. With **TYPE=VOLUME +HIDDEN**, facets which are shared by volumes with the same labels are omitted.

Hidden surface pictures can only be displayed after the surface mesh has been calculated (see “[The MESH Command](#)” on page 4-133).

The discretisation of the surfaces and volumes can also be displayed using the **ELEMENT** parameter. If the volume or surface meshes have not been calculated, the program displays using the nearest options it can.

Hidden surface views of **FACETS**, with **LABEL** set to a single potential name (**AX**, **AY**, **AZ**, **POTENTIAL**, **VOLTAGE** or the incident potentials) are displayed as contour plots to show the values of the potential on the facets.

Arrows to show the direction of the volume vector properties can be drawn inside each volume. By default, the arrows are only drawn inside the conductors to show the current direction (**VECTOR=CONDUCTORS**) but they can also show the

**MATERIAL** orientation, the **CURRENT** density direction or the **VELOCITY** direction. The arrow display can be switched off completely (**VECTOR=NO**).

## Examples

The following example **DISPLAY** commands assume program default values, and that the commands are issued in sequence. The model is assumed to occupy a unit cube in the positive **X**, **Y** and **Z** octant of the coordinate system.

- Example: to display a wire-frame picture without conductors, orthogonal projection from the **Z** direction (note the use of positional parameters):

```
OPERA > display 0.5 xorigin=0.5 0.5 0.5 -coil
```

- Example: to display the conductors, and check that the reduced potential volumes enclose them:

```
OPERA > xeye=3 4 5 +coil labe=reduced
```

- Example: to obtain a “hidden-surface” picture of the conductors and one material, checking the finite element discretisation on the surfaces of the volumes, and using perspective (note that the eye position has to be moved further away to reduce the affect of the perspective):

```
OPERA > xeye=9 12 15 +perspective label=iron,
```

```
OPERA > +hidden, element=surface
```

- Example: to display the surfaces with the **NORMAL MAGNETIC** boundary condition:

```
OPERA > type=face label=normmagn
```

## The **DUMP** Command

---

### Menu Route:

OPTIONS ↓  
Dump picture

### Command Line Parameters:

Command	<b>DUMP</b>	
Parameter	Default	Function
<b>FILE</b>	none	Name of file to contain the picture.
<b>TYPE</b>	<b>POSTSCRIPT</b>	Graphics language:
	<b>HPGL</b>	Hewlett-Packard Graphics Language
	<b>PICOUT</b>	Vector Fields PICOUT Graphics Language
	<b>POSTSCRIPT</b>	Adobe PostScript
<b>SIZE</b>	<b>A4</b>	Paper sizes: HPGL can use <b>A, A3, A4, B, A0D, A1D, A2D, A3D</b> or <b>A4D</b> ; PostScript can use <b>A4, A</b> or <b>USER</b> .
<b>LLX</b>	0	X-coordinate of lower-left corner in mm ( <b>TYPE=POST, SIZE=USER</b> ).
<b>LLY</b>	0	Y-coordinate of lower-left corner in mm ( <b>TYPE=POST, SIZE=USER</b> ).
<b>URX</b>	0	X-coordinate of upper-right corner in mm ( <b>TYPE=POST, SIZE=USER</b> ).
<b>URY</b>	0	Y-coordinate of upper-right corner in mm ( <b>TYPE=POST, SIZE=USER</b> ).
<b>COLOUR</b>	<b>YES</b>	Colour PostScript: <b>NO</b> implies grey-scale.
<b>FILL</b>	<b>NO</b>	Filled polygons in HPGL: <b>YES</b> or <b>NO</b> .
<b>ORIENT</b>	<b>LANDSCAPE</b>	PostScript paper orientation: <b>LANDSCAPE</b> or <b>PORTRAIT</b> .
<b>SWAP</b>	<b>YES</b>	Swap black and white in PostScript:
	<b>NO</b>	Colours appear as on screen.
	<b>YES</b>	Black and white are swapped.

The **DUMP** command copies the graphics commands used to create the current display to a **FILE** in three different formats. If no filename extension is given, extensions *.ps*, *.hgl* or *.pic* are added as appropriate. Note that the 3D Viewer window cannot be copied to a file using the **DUMP** command. This window must be saved as a bitmap directly.

- Adobe PostScript (**TYPE=POST**): This has options for paper size, colour and orientation.
  - There are two standard paper sizes: European **A4** and American **A**. Alternatively **SIZE=USER** allows the image to be printed at any size and any position on the page by the specification of the coordinates of the lower-left and upper-right corners in mm (parameter **LLX**, **LLY**, **URX**, **URY**).
  - If **+COLOUR** is selected, the screen colours will be matched on the paper as closely as possible, except that black and white can be optionally **SWAPPED**. (Black is any colour with an intensity of less than  $\frac{1}{256}$  for Red, Green and Blue; white is any colour with an intensity of more than  $\frac{255}{256}$  for Red, Green and Blue – see the **COLOUR** command.)
  - Grey-scale pictures can be created in several ways: the colours could be changed to grey using the **COLOUR** command before **DUMPING** (see the **COLOUR** command); a colour picture could be sent to a grey-scale printer; or a grey-scale picture could be stored using **-COLOUR**. In this last case, the grey levels are calculated using the formula  $1 - \frac{r + g + b}{3}$ . Thus dark colours on the screen become light colours on the paper.
  - Pictures can be orientated in two ways: **PORTRAIT** pictures have the X horizontal along the bottom of the page and the vertical axis up the left-hand side; **LANDSCAPE** pictures have the horizontal axis up the right-hand side of the page and the vertical axis from right to left along the bottom of the page.
- HPGL (**TYPE=HPGL**) pictures are intended for pen-plotters. There is a larger selection of sizes available: the sizes with **D** appended are for drum plotters, the other sizes are for flat-bed plotters.
  - The only other option allows polygon-fill to be selected (**± FILL**). If selected, polygons are filled using shading with parallel horizontal or vertical lines.
- PICOUT (**TYPE=PICOUT**) pictures can be re-displayed or translated using the PICOUT program supplied by Vector Fields for OpenVMS and UNIX systems. It reads graphics files and redisplayes them on the screen or converts them to PostScript or HPGL for printing or plotting. The PICOUT program prompts

the user for the name of the input file and (if necessary) the name of the output file. It also prompts for options such as paper size, colour or monochrome, etc., although in some implementations these can be supplied by command line options (*-cc*) which are shown below.

Instructions for running PICOUT are included in the Implementation Notes supplied with the software.

PICOUT prompts the user for the input it needs. On UNIX systems, some choices can be specified as command line options, for example *-sc* below. Replies to PICOUT's prompts can be abbreviated.

PICOUT always prompts for the name of an input file. There are 3 output options: Screen (*-sc*), PostScript (*-ps*) and HPGL (*-hp*).

- **Screen:** If output to the screen is chosen, there are no further options. On workstations and X-terminals, after the graphics window has been created, its size can be adjusted using the window manager and the pictures will be scaled to fit the new size. The original aspect ratio of the pictures will be maintained by increasing either the top or right side margin. PICOUT pauses for *<carriage-return>* to continue or *Q* to quit at the end of each picture.
- **PostScript:** If PostScript output is chosen, PICOUT prompts for the name of the file to contain the PostScript program. If the file already exists it will be overwritten. When PICOUT has finished, this file can be sent directly to a PostScript printer. PostScript output can be written to multiple files with a single page in each file (*-s*) or to one file with many pages in the file (*-m*). The advantage of multiple files is that a single page can be printed without printing all the others. The filenames for the single page, multiple file option are formed by appending *.1*, *.2*, *.3* etc. to the name given at the output file prompt. On some operating systems, it is necessary to name the output file with a single part name in order to make use of this facility.

PostScript output can be created for different paper sizes. There are 2 standard sizes: European A4 (*-a4*) and American A (*-a*) for which PICOUT knows the paper size. Non-standard user defined paper sizes (*-u llx lly urx ury*) can also be given by specifying the coordinates of the lower left and upper right corners of the drawing area in millimetres.

PostScript output can be orientated on the page in landscape format (*-l*) or portrait format (*-p*). For paper sizes A and A4, portrait format fits 2 pictures on each page.

PostScript output can be in colour (*-c*) or greyscale (*-g*). Colour output assumes white paper, so the black and white of the original screen picture are swapped. Greyscale intensities are inverted so that dark colours appear pale grey and bright colours as dark grey. Some greyscale printer

can interpret colour commands and therefore give a choice of using PICOUT's algorithm for colour to greyscale conversion or the printer's algorithm.

- **HPGL:** If HPGL output is chosen, PICOUT prompts for the name of the file to contain the HPGL commands. Each page of output is written to a separate file. The filenames are formed by appending .1, .2, .3 etc. to the name given at the output file prompt. If a file already exists it will be overwritten. When PICOUT has finished, the files can be sent directly to a HPGL printer or plotter. On some operating systems, it is necessary to name the output file with a simple name so that the extension can be added.

HPGL output can be created for a variety of paper sizes. For flat-bed plotters, sizes A (**-a**), A3 (**-a3**), A4 (**-a4**) and B (**-b**) are available. For drum plotters, HPGL output can be produced for sizes A0D (**-a0d**), A1D (**-a1d**), A2D (**-a2d**), A3D (**-a3d**) and A4D (**-a4d**). The difference between the flat-bed and drum sizes is that the coordinate system origin is at the centre of the plot on drum plotters and at the bottom left corner on flat bed plotters. No PS (paper size) command is put into the files.

HPGL output can include extra line drawing commands to shade filled polygons (**-f**) or polygons can be represented by the outlines alone (**-nf**).

## PICOUT OPTIONS

Output style:

- sc Screen. PICOUT redraws the pictures onto the workstation screen.
- ps PostScript. PICOUT creates a file or files of PostScript commands.
- hp HPGL. PICOUT creates files of HPGL commands.

Paper size (ignored with **-sc**):

- size PostScript output can be produced for standard paper sizes A (**-a**) and A4 (**-a4**) or for a user choice of paper size, (**-ullx lly urx ury**). The integer numbers given by *llx*, *lly*, *urx* and *ury* specify the position of the lower left and upper right corners of the picture in millimetres.
- size[d] For flat-bed plotters, HPGL output can be produced for sizes **a**, **a4**, **a3** and **b**; for drum plotters, HPGL output can be produced for sizes **a0d**, **a1d**, **a2d**, **a3d** and **a4d** (**d** for 'drum'.)

Polygon fill (HPGL only):

**PICOUT OPTIONS** (*continued*)

- f Fill polygons (solid shading).
- nf Do not fill polygons (outlines drawn).

## Colour or greyscale (PostScript only):

- c Colour. White paper is assumed, so black and white are swapped.
- g Greyscale. Intensities are inverted, so that dark colours appear as pale grey and bright colours as dark grey.

## Picture orientation (PostScript only):

- l Landscape (one picture on each page).
- p Portrait (two half-size pictures on each page).

## File output (PostScript only):

- s Single page per file. Each picture is written to a separate file, with filenames generated from the output file name given by the user.
- m All pages in one file.

## Help:

- h For help (all other options ignored).

## The **EDIT** Command

---

Not available from Menus

### Command Line Parameters:

Command	<b>EDIT</b>	
Parameter	Default	Function
<b>FILE</b>	<i>none</i>	Name of OPERA-3d pre processor data file.

The **EDIT** command reads an OPERA-3d pre processor data file into the program in edit mode. This allows major changes to be made to the data, including changes to the topology of the base plane. The data already stored in the pre processor can also be changed using the **EXTEND** command (page 4-113), the **MODIFY** command (page 4-135) and the **REDEFINE** command (page 4-151).

The **EDIT** command has one parameter which defines the name of the **FILE**. If no file name extension is given, the extension *oppre* is assumed.

As the file is being read a certain amount of editing can be done. Each top-level command can be **EXECUTED**, **IGNORED** or replaced; and at breakpoints within the **DEFINE** command the user can add additional information or change the data which has been read. At any point it is possible to **FINISH** editing and read the rest of the file or **SKIP** over the rest of the file and return immediately to normal control.

All the commands which the **EDITOR** executes are decoded, so that the parameter values are stored but only certain commands are obeyed. These are the commands which define finite element data (**DEFINE**, **EXTEND** and **MODIFY**), conductor data (**CONDUCTOR**) and built-in commands (**\$ OS** and **\$ CD** are not executed).

More details of reading OPERA-3d pre processor data files are given with the **READ** command. When the editing is complete, the **WRITE** command can be used to store the edited data in a file.

The **EDIT** command should not be used in a **\$ COMINPUT** file.

## Editing Top-level Commands

As the program reads each top-level command from the file, the command is displayed on the terminal with the message

Your next input was:

and the user has the opportunity to **EXECUTE**, **IGNORE** or replace it, to **SKIP** over all of the remaining commands in the file or to **FINISH** reading and executing until the bottom of the file is reached.

Replacement commands should not introduce extra interactions with the program, since it is not possible to insert new command lines.

Top-level Edits	
Keyword	Meaning
<b>FINISH</b>	Continue reading to end of file.
<b>IGNORE</b>	Ignore the displayed command.
<b>SKIP</b>	Ignore the rest of the file.
<b>XECUTE</b>	Execute the displayed command.
<i>replacement command</i>	Execute the replacement command.

## Editing the DEFINE Command

As the sub-commands and cursor-hits within the **DEFINE** command are being read, some of the keyboard command lines can be edited in the same way as top-level commands above. It is not usually sensible to use the **IGNORE** option except to remove an invalid command line from the file otherwise the sequence of commands would be destroyed. Similarly, a replacement command should not invoke extra interaction with the program, since it is impossible to add extra command lines.

Edits to <b>DEFINE</b> sub-commands	
Keyword	Meaning
<b>FINISH</b>	Continue reading to end of file.
<b>IGNORE</b>	Ignore the displayed command.
<b>SKIP</b>	Ignore the rest of the file.
<b>XECUTE</b>	Execute the displayed command.
<i>replacement command</i>	Execute the replacement command.

Additional break points are included at the ends of each of the modes of the **DEFINE** command. At these break points a question requiring the answer **YES** or **NO** is given enabling the user to define more points or facets, redefine facet

subdivisions, move points on subsequent planes or redefine materials and boundary conditions.

Note that there is only a break point at the ends of Point Definition Mode and Facet Definition Mode when those modes are left for the first time. There is no break point when Point Definition Mode is re-entered from Facet Definition Mode or when Facet Definition Mode is re-entered from Subdivision Mode. However it is possible to re-enter Point Definition Mode from the break point at the end of Facet Definition Mode and to re-enter Facet Definition Mode from the break point at the end of Subdivision Mode.

- Example - at the end of Base Plane Facet Definition Mode, the message given is:

```
End of facet definition mode. Do you need to define
more facets? (Y or N)
```

In response to a **YES** reply, the program enters the appropriate **DEFINE** command mode as described on page 4-49. It is usually necessary to use the **R** cursor hit first so that the current state of the data can be displayed. When a **Q** cursor hit is used to leave the mode, the program resumes reading the file, until the next break point is reached. In the case of the above example, a **YES** response makes the Facet Definition Mode cursor commands available, including **N** for returning to Point Definition Mode (page 4-55).

**FINISH** and **SKIP** can be used as replies instead of **YES** or **NO** to end the editing by reading and executing the rest of the file or returning immediately to normal input.

Edits at <b>DEFINE</b> break-points	
Keyword	Meaning
<b>FINISH</b>	Continue reading to end of file.
<b>NO</b>	Continue reading to the next break point.
<b>SKIP</b>	Ignore the rest of the file.
<b>YES</b>	Make cursor input available to edit the data.

## The **END** Command

---

### *Menu Route:*

FILE↓  
End OPERA-3d/Pre

### *Command Line Parameters:*

Command	<b>END</b>
No Parameters	

The **END** command stops the OPERA-3d pre processor. All data files are closed.

Note that it is important to **WRITE** a pre processor data file ([page 4-179](#)) before ending the program so that all the commands and data are recorded.

If the program is ended without a **WRITE** command having been issued to store the pre processor data, then the file *Opera3d\_Pre\_n.backup* can be renamed to have a file name extension *oppre*. This file is equivalent to a pre processor data file.

## The **EXTEND** Command

---

### Menu Route:

DEFINE↓  
Extend existing mesh

### Command Line Parameters:

Command	<b>EXTEND</b>			
Parameter	Default	Function		
<b>MESH</b>	1	Number of finite element mesh to be extended.		
<b>EDIT</b>	<b>YES</b>	Material and boundary condition editing switch:		
		<table border="0"> <tr> <td><b>NO</b></td> <td>Materials left as <b>AIR TOTAL LINEAR</b>, no boundary conditions set.</td> </tr> <tr> <td><b>YES</b></td> <td>Material properties and boundary conditions can be set after extensions.</td> </tr> </table>	<b>NO</b>	Materials left as <b>AIR TOTAL LINEAR</b> , no boundary conditions set.
<b>NO</b>	Materials left as <b>AIR TOTAL LINEAR</b> , no boundary conditions set.			
<b>YES</b>	Material properties and boundary conditions can be set after extensions.			

The **EXTEND** command puts the pre processor into its finite element mesh creation mode but starts from the top plane of an existing mesh. The user input is tightly structured by the program. The top mesh plane surface is extruded or swept through space thus creating new layers of volumes. The topology is maintained during the extrusion operations, but the point coordinates can be changed in the new surface created by each extrusion or sweep. If editing has been selected (**+EDIT**), once all the new layers have been created, the volumes in the new layers of the mesh can be assigned material attributes and boundary conditions can be assigned to the faces of the volumes.

*In menu mode* the **Extend with editing** option with editing adds one additional extrusion at a time and allows coordinates, material properties and boundary conditions to be edited. **Extend without editing** can be used to add several extrusions, but any variation in coordinate positions, material properties and boundary conditions must be applied later using **MODIFY**.

The **MESH** parameter specifies the number of the finite element mesh to be extended.

### Extrusions Mode

The **EXTEND** command applies extrusion operations to the set of facets of the top mesh plane of previously created mesh. There must be at least one extrusion

operation, but many others may be needed to define the complete problem. It is also possible to add more extrusions to a completed mesh using the **EXTEND** command again. Point movements and transformations can be performed in the same way as for **DEFINE**. The details of the command lines to define the extrusions are given with the **DEFINE** command (page 4-49).

After the extrusion has been created and all necessary points have been moved, the program moves on to the next extrusion. The user is given the option of creating another extrusion or not (valid replies are **YES** or **NO**). After a **NO** response, the program moves on to Material Definition Mode for the new layers.

## Material Definition Mode

It is now necessary to set the material and mesh properties within each new volume. This is done by presenting the user with each layer in turn and the user setting values which over-ride the default or current settings within each volume. Only the new layers are presented to the user, but lower numbered layers can be modified by using the **FROM** keyword on the material definition command line.

Volumes are selected and materials and properties defined in the same ways as for the **DEFINE** command (page 4-49). When all necessary changes have been made the program moves on to Boundary Condition Definition Mode for the new facets, starting with the mesh plane which was previously the top plane.

## Boundary Condition Definition Mode

Boundary conditions can be set on any surface of any of the new volumes in the mesh. In order to achieve this the program presents the facets in 3 separate groups: first the facets on the planes from the previous top plane to the one below the new top plane, second the final plane of the mesh, and last the facets normal to the planes, for each of the new layers, one layer at a time. Facets of lower numbered planes or layers can be modified by using the **FROM** keyword on the boundary condition definition command line.

Facets are selected and boundary conditions defined in the same ways as for the **DEFINE** command (page 4-49). When all necessary changes have been made the program ends the **EXTEND** command and waits for another top-level command.

## The **FILL** Command

---

### Menu Routes:

```
MESH↓
  Volume mesh ...
  ... options
  Mesh
```

### Command Line Parameters:

Command	<b>FILL</b>	
Parameter	Default	Function
<b>TOLERANCE</b>	1.E-5	Geometric Tolerance
<b>PRINT</b>	0	Diagnostic printing level (0 or 1)

The **FILL** command generates the volume finite element mesh in all the pre processor volumes. The type of mesh (tetrahedra or hexahedra) is determined by which type of surface mesh has been created by the **MESH** command ([page 4-133](#)).

### Tetrahedral meshes

Each region of the model is meshed independently, given the required triangular element mesh on the surfaces that has been created by the **MESH** command. If the volume has only 3 or 4 sided facets with regular subdivisions, the internal nodes are regularly positioned. Otherwise, the meshing is based on Delaunay point insertion, followed by maximising the minimum angle of the elements.

The **TOLERANCE** parameter may need to be adjusted if the software fails to create a mesh successfully. Problems can occur if the element size varies too much between the surfaces of a region; the tolerance should be increased to overcome this. In other cases, for example, a complicated region with a wide range of feature sizes, the tolerance may need to be reduced.

If the mesh generation process fails, and the **TOLERANCE** parameter does not correct the problem, then it is likely that the element size is changing too much between adjoining surfaces. Regions that cannot be meshed are flagged using the **DEBUG** label and can be selected for **DISPLAY**. Adjust the edge subdivisions to reduce the element size variations in such regions.

## Hexahedral meshes

Hexahedral elements can only be generated if the model uses regions that are hexahedra or degenerate hexahedra. A regular finite element mesh is generated in all regions, by subdivision of the region to similar shaped elements. The **TOLERANCE** parameter is used to test for coincident points in degenerate shapes.

## The **HELP** Command

---

### Menu Route:

HELP↓  
Help

### Command Line Parameters:

Command	<b>HELP</b>
No Parameters	

The **HELP** command gives help to remind users of several of the features of the program. The topics are:

- **System Overview:** this gives a flow-chart of the OPERA-3d pre processor top level commands indicating the sequence in which they should be used to prepare a data set for analysis.
- **Command Interpreter:** this summarizes the syntax and built-in help features of the command decoder, including details of sub-commands and cursor commands. More information is given in chapter 2
- **Euler Angles:** the definitions of the Euler Angles used within OPERA-3d is given, and information on how to use the escape function **\$EULER** to specify them. More information is given on [page 2-31](#)
- **New Features:** this summarize the features of the program which have been added since the previous version. **Units:** this lists the units used in the systems allowed by the analysis programs.

The program prompts for the name of a topic.

Keyword	Meaning
<b>COMMAND</b>	Command interpreter.
<b>EULER</b>	Euler angles.
<b>NEW</b>	New features.
<b>SYSTEM</b>	System overview.
<b>UNITS</b>	Unit systems.
<b>QUIT</b>	Leave the <b>HELP COMMAND</b> .

When accessed from the menus, the **HELP** command has different topics explaining the use of the program and the menu interface.

## The IDEAS Command

---

### Menu Route:

FILE↓  
Read I-DEAS universal file

### Command Line Parameters:

Command	IDEAS
No Parameters	

The **IDEAS** command introduces a set of sub-commands which can be used to read a finite element mesh from an I-DEAS Universal File and associate OPERA-3d material properties and boundary conditions with the elements and nodes it contains.

The **READ** sub-command is used to read the data. Additional information about materials and boundary conditions can be supplied using the sub-commands **BOUNDARY** and **MATERIAL** after the file has been read. Conductors can be added and the data can be **DISPLAYED** (page 4-97) and written to an OPERA-3d analysis database (see “The **SOLVERS** Command” on page 4-153), but the finite element data cannot be modified.

### Universal Data File for Input to OPERA-3d

The Universal Data File created by I-DEAS must contain the following Datasets (Level 6 or Master Series):

Dataset Name	Dataset Number
Header	151
Nodes	15, 781 or 2411
Material Properties	91, 747, 773 or 1710
Elements	71, 780 or 2412
Restraints or Loads	755, 756, 782, 791 or 792

The order of the datasets is not important except that any data which is referenced by other datasets must be defined before it is referenced; e.g. nodes and materials must be defined before elements. Other datasets may also be included in the input file, but will be ignored.

The I-DEAS Universal Data File Level 6 and Master Series formats are defined in the I-DEAS documentation published by SDRC, and are not repeated here. The following information describes how the data is used in interface to OPERA-3d.

### *Material Properties*

Up to 100 materials may be used in the universal file. The properties in the universal file are ignored; only the material number and name are used by the interface. A different material should be used for each different combination of OPERA-3d material and volume properties. For example, in the universal file data, the volume representing free space might be divided into two parts with different material names, so that in OPERA-3d part can use **REDUCED** scalar potential and the other part **TOTAL** potential. Each material defined in the universal file will initially be given the properties of **AIR** with **TOTAL** scalar potential and **LINEAR** elements. The OPERA-3d material name, potential type and element type can be changed and other volume properties added using the **MATERIAL** sub-command (page 4-122).

### *Element Topologies*

The interface is able to process elements of the following types:

111	Linear tetrahedron
115	Linear hexahedron
116	Quadratic hexahedron
118 or 119	Quadratic tetrahedron

In any universal file, only one topology of element (tetrahedron or hexahedron) can be used; mixed element types are not allowed.

All quadratic elements in the universal file are converted to linear elements by the interface. Quadratic elements can be selected either by specifying that a particular universal file material should use quadratic elements (**MATERIAL** sub-command, page 4-122) or that all elements should be quadratic when the analysis database is created (**SOLVERS** command, page 4-153).

### *Restraints and Loads*

Restraint and Load datasets identify the sets of nodes which should have the same boundary condition. The **BOUNDARY** sub-command can be used to associate one or more OPERA-3d boundary conditions with each set. The types and values of the boundary conditions in the universal file are ignored. Only the node numbers are used.

## Displaying Universal File Data

Universal File data can be displayed using the **DISPLAY** command (page 4-97). Line drawings are not available, but displays without hidden surfaces can show surface or volume elements.

As the universal file data is being read, the elements are allocated to volumes, one for each universal file material. A label **MATE $n$**  (where  $n$  is the universal file material number) is added to each volume so that they can be selected for display. After OPERA-3d properties have been added, the material names and other property labels can be used as well.

Similarly, the element facets identified by the nodes in each boundary condition set are grouped and given facet labels **BCSET $n$**  (where  $n$  is a counter of how many boundary condition sets have been read). The boundary condition facets can be displayed using these labels. After the OPERA-3d boundary conditions have been applied using the **BOUNDARY**, the OPERA-3d boundary condition names can be used as well.

## The IDEAS Sub-command **BOUNDARY**

### Menu Route:

**FILE**↓  
I-DEAS universal file → Define Boundary Conditions

### Command Line Parameters:

Sub-command	<b>BOUNDARY</b>	
Parameter	Default	Function
<b>NUMBER</b>	<i>none</i>	Boundary condition set number
<b>OPTION</b>	<b>ENQUIRE</b>	Option:
	<b>ENQUIRE</b>	List boundary conditions defined for this set.
	<b>SET</b>	Define or clear boundary conditions for this set.

Sub-command	<b>BOUNDARY</b> (continued)	
Parameter	Default	Function
<b>CONDITION</b>	<i>none</i>	Boundary condition name or <b>NONE</b> :
		<b>AX</b> <b>AX</b> boundary condition.
		<b>AY</b> <b>AY</b> boundary condition.
		<b>AZ</b> <b>AZ</b> boundary condition.
		<b>DAX</b> <b>DAX</b> boundary condition.
		<b>DAY</b> <b>DAY</b> boundary condition.
		<b>DAZ</b> <b>DAZ</b> boundary condition.
		<b>DERIVATIVE</b> <b>DERIVATIVE</b> boundary condition.
		<b>INAX</b> <b>INAX</b> boundary condition.
		<b>INAY</b> <b>INAY</b> boundary condition.
		<b>INAZ</b> <b>INAZ</b> boundary condition.
		<b>NONE</b> Clear all boundary conditions.
		<b>NORMELEC</b> <b>NORMAL ELECTRIC</b> boundary condition.
		<b>NORMMAGN</b> <b>NORMAL MAGNETIC</b> boundary condition.
		<b>PEC</b> <b>PEC</b> boundary condition.
		<b>POTENTIAL</b> <b>POTENTIAL</b> boundary condition.
		<b>SYMMETRY</b> <b>SYMMETRY</b> boundary condition.
		<b>RADIATION</b> <b>RADIATION</b> boundary condition.
		<b>SLIP</b> <b>SLIP</b> boundary condition.
		<b>SYMMETRY</b> <b>SYMMETRY</b> boundary condition.
		<b>TANGELEC</b> <b>TANGENTIAL ELECTRIC</b> boundary condition.
		<b>TANGMAGN</b> <b>TANGENTIAL MAGNETIC</b> boundary condition.
		<b>VOLTAGE</b> <b>VOLTAGE</b> boundary condition.
<b>VALUE</b>	<i>none</i>	Potential value.
<b>LABEL</b>	<i>none</i>	Drive label.

The **BOUNDARY** sub-command can be used to list existing boundary conditions on the set of nodes given by **NUMBER**, (**OPTION=ENQUIRE**) or add new ones (**OPTION=SET**, **CONDITION=name**). **OPTION=SET**, **CONDITION=NONE** clears all existing conditions on the set of nodes. The PERA-3d boundary conditions are described on [page 4-86](#).

## The IDEAS Sub-command MATERIAL

### Menu Route:

FILE↓

I-DEAS universal file → Define Materials

### Command Line Parameters:

Sub-command	MATERIAL	
Parameter	Default	Function
<b>NUMBER</b>	<i>none</i>	Universal file material number.
<b>OPTION</b>	<b>ENQUIRE</b>	Option:
	<b>ENQUIRE</b>	List material properties.
	<b>SET</b>	Define material properties.
<b>NAME</b>	<i>none</i>	OPERA-3d material name.
<b>POTENTIAL</b>	<i>none</i>	Potential type:
	<b>REDUCED</b>	Reduced magnetic scalar potential.
	<b>TOTAL</b>	Total scalar potential.
	<b>VECTOR</b>	Magnetic vector potential.
<b>ELEMENT</b>	<i>none</i>	Element type:
	<b>LINEAR</b>	First order elements.
	<b>QUADRATIC</b>	Second order elements.
<b>PACK</b>	<i>none</i>	Packing factor.
<b>SCALAR</b>	<i>none</i>	Scalar property: charge density or rotational velocity.
<b>CURX</b>	<i>none</i>	X-component of current density.
<b>CURY</b>	<i>none</i>	Y-component of current density.
<b>CURZ</b>	<i>none</i>	Z-component of current density.
<b>VELX</b>	<i>none</i>	X-component of velocity.
<b>VELY</b>	<i>none</i>	Y-component of velocity.
<b>VELZ</b>	<i>none</i>	Z-component of velocity.
<b>THETA</b>	<i>none</i>	Material orientation Euler Angle $\theta$ .

Sub-command	<b>MATERIAL</b> ( <i>continued</i> )	
Parameter	Default	Function
<b>PHI</b>	<i>none</i>	Material orientation Euler Angle $\phi$ .
<b>PSI</b>	<i>none</i>	Material orientation Euler Angle $\psi$ .

The **MATERIAL** sub-command can be used to list existing material properties (**OPTION=ENQUIRE**) or define new ones (**OPTION=SET**). **OPTION=ENQUIRE** also sets the default values of the other parameters to the current values for the material.

The current density, velocity and material orientation vectors can be expressed as functions of **X**, **Y** and **Z**.

The OPERA-3d material properties are described in the section “**Material Definition Mode**” on page 4-78.

## The **IDEAS** Sub-command **QUIT**

### *Menu Route:*

**FILE**↓  
I-DEAS universal file → Return

### *Command Line Parameters:*

Sub-command	<b>QUIT</b>
No Parameters	

The **QUIT** sub-command leaves the **IDEAS** command and returns to the top-level commands.

## The IDEAS Sub-command READ

### Menu Route:

FILE↓

I-DEAS universal file → Read File

### Command Line Parameters:

Sub-command	READ	
Parameter	Default	Function
FILE	<i>none</i>	Name of file.

The **READ** sub-command reads a Universal File to input materials, nodes, elements and boundary conditions. If no file name extension is given, the extension *unv* is assumed.

## The LABEL Command

### Menu Route:

MODIFY↓  
Labels

### Command Line Parameters:

Command	LABEL	
Parameter	Default	Function
MESH	1	Number of finite element mesh to be labelled.
START	1	Number of the first plane or layer. Planes are numbered: 1,2, 3, etc.; mid-extrusion planes and layers are numbered: 1.5, 2.5, 3.5, etc.

The LABEL command allows labels to be added or removed from the entities (points, lines, facets and volumes) of the finite element mesh. Some labels are added automatically by the DEFINE command (page 4-49): every entity has label ALL, facets have the boundary condition types, and volumes have the material names, potential types and element types. Such labels can only be changed by the MODIFY command. Other labels are added to parts of the model by the CHECK command (page 4-18), FILL command (page 4-115), MESH command (page 4-133) and SLIP command (page 4-152). Additional labelling by the LABEL command allows entities to be grouped for DISPLAY and points to grouped for the TRANSFORM command (page 4-178). To enable multiple points to be labelled more easily, any labelling operation on volumes, facets or lines is also applied to the points which define them.

Facet and volume labels are stored in analysis databases and can be used in the post processor.

The data of the finite element mesh is separated into 4 sections, corresponding to the modes of the DEFINE command. The user is asked first to choose between points, lines, facets and volumes.

LABEL Modes	
Keyword	Meaning
FACETS	Add or remove labels on facets.
LINES	Add or remove labels on lines.
POINTS	Add or remove labels on points.

LABEL Modes (continued)	
Keyword	Meaning
QUIT	Leave the LABEL command.
VOLUMES	Add or remove labels on volumes.

Valid replies are: **FACETS**, **LINES**, **POINTS**, **QUIT** or **VOLUMES**.

*In menu mode* labels can be set on one plane of layer at a time. The program prompts for a value of **Plane number** or **Layer number**. The menu item **Select/de-select entity** allows entities to be added or removed from a list. Menu item **Select and define** adds one last entity to the list and causes the program to display a DialogBox into which the labels can be entered. The menu item **List conditions** allows the labels to be listed for an individual entity.

*In keyboard mode*, each mesh plane or extrusion layer of the mesh specified by the **MESH** parameter, starting with the plane or layer given by the **START** parameter, is presented to the user. Mesh planes are numbered 1, 2, 3, etc., and mid-extrusion planes and layers 1.5, 2.5, 3.5, etc. Layer 1.5 is between plane 1 and plane 2. The facets are presented in two sections: first those lying in the mesh planes, then those connecting the extruded planes. Entities can be selected using the cursor, singly with a **<space>**, or in groups with **K** for all but the last and **<space>** for the last.

A new label can be **ADDED** to or an existing label **REMOVED** from the selected entities and corresponding entities in other planes or layers. For each set of entities the selection cursor hits and labelling sub-commands are the same.

## Entity Selection

Entity Selection Menu and Cursor Hits		
Cursor hit	Menu item	Function
<b>&lt;space&gt;</b>	Select and define	Select entity nearest cursor and prompt for a labelling command.
<b>F</b>	Finish	Finish labelling this type of entity.
<b>H</b>		Display menu help message explaining all the cursor options. (More help is available after entity selection.)

Entity Selection Menu and Cursor Hits ( <i>continued</i> )			
Cursor hit	Menu item	Function	
<b>K</b>	Select/de-select entity	Select the entity nearest the cursor. Repeating <b>K</b> for a selected entity, de-selects it. A labelling command for all the selected entities can be given after a <b>&lt;SPACE&gt;</b> cursor hit.	
<b>L</b>	List labels	List all the labels on the nearest entity.	
<b>Q</b>		Leave this plane or layer.	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
	<b>&lt;return&gt;</b>	Reconstruct at the same size.	
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.	
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off ( <a href="#">page 4-52</a> ).	

## Labelling Sub-commands in Keyboard Mode

Following a <space> cursor hit, one of the following sub-commands should be given. The commands apply to the entity selected with the <space> and to any previously selected with **K**.

The labelling sub-commands consist of the following items:

Labelling Sub-commands	
Command and parameter:	
<b>ADD</b> <i>label</i>	Add a label to the selected entities.
<b>HELP</b>	Obtain help on all sub-commands and options.
<b>QUIT</b>	Abandon the currently selected entities.
<b>REMOVE</b> <i>label</i>	Remove a label to the selected entities.
Optional keywords - setting multiple entities:	
<b>ALL</b>	Sets all entities in layer(s) or plane(s).
<b>FROM</b> <i>value</i>	Sets all layers/planes from layer/plane number given through to the current layer/plane or the layer/plane number given with <b>TO</b> .
<b>KEEP</b>	Keeps current selection of entities for another <b>ADD</b> or <b>REMOVE</b> command.
<b>TO</b> <i>value</i>	Sets all layers/planes from current layer/plane or the layer/plane number given with <b>FROM</b> through to the layer/plane number given. The value may be * to indicate the top layer/plane.

## Labelling DialogBox in Menu Mode

In the DialogBox, **ADD** or **REMOVE** must be selected and a label name given.

Label Definition

Operation:

Add a label     Remove a label

Label name

From  To   All

## The **MATERIALS** Command

---

*Menu Route:*      DEFINE↓  
                           Material Properties

*Command Line  
 Parameters:*

Command	<b>MATERIALS</b>		
Parameter	Default	Function	
<b>OPTION</b>		<b>PICK</b>	Adds a material label to a list to be set.
		<b>UNPICK</b>	Clears the list of material labels to be set.
		<b>RESET</b>	Sets picked materials to have the properties of air.
		<b>MODIFY</b>	Sets the data for the picked materials.
		<b>METRE</b>	Work in SI units.
		<b>CGS</b>	Work in CGS units.
		<b>INCH</b>	Work in SI units with inches.
		<b>MM</b>	Work in SI units with MM.
		<b>MICRON</b>	Work in SI units with microns.
		<b>LIST</b>	Lists the material properties of the picked materials.
	<b>DELETE</b>	Deletes the picked materials.	
<b>MATERIALLABEL</b>		Material label used with <b>OPTION=PICK</b>	
<b>ANISOTROPY</b>		Sets the anisotropy of all permeability, permittivity and conductivity	
		<b>ISOTROPIC</b>	Set to isotropic.
		<b>PACKED</b>	Set to packed.
	<b>ANISOTROPIC</b>	Set to anisotropic.	

Command	MATERIALS <i>(continued)</i>	
Parameter	Default	Function
LINEARITY		Sets the properties to be linear or non-linear.
	LINEAR	Set to linear.
	NONLINEAR	Set to non-linear.
MUANISOTROPY		Sets the anisotropy of permeability.
	ISOTROPIC	Set to isotropic.
	PACKED	Set to packed.
	ANISOTROPIC	Set to anisotropic.
MULINEARITY		LINEAR
		NONLINEAR
MU		Isotropic linear permeability.
HC		Isotropic coercivity.
BH		Isotropic non-linear BH curve.
MPHASE		Phase lag for isotropic permeability.
MUXX		Anisotropic components of linear permeability.
MUY Y		
MUZZ		
HCX		Anisotropic components of linear coercivity.
HCY		
HCZ		
BHX		Anisotropic components of non-linear permeability.
BHY		
BHZ		
MAPHASE		Complex phase lag for anisotropic permeability.
SIGANISOTROPY		Sets the anisotropy of conductivity.
	ISOTROPIC	Set to isotropic.
	ANISOTROPIC	Set to anisotropic.
SIGMA		Isotropic conductivity.
SPHASE		Phase lag for isotropic conductivity.
SIGXX		Anisotropic components of conductivity.
SIGYY		
SIGZZ		
SAPHASE		Complex phase lag for anisotropic permittivity.
EPSANISOTROPY		Sets the anisotropy of permittivity.
	ISOTROPIC	Set to isotropic.
	ANISOTROPIC	Set to anisotropic.

Command	<b>MATERIALS</b> ( <i>continued</i> )	
Parameter	Default	Function
<b>EPSILON</b>		Phase lag for isotropic permittivity.
<b>EPHASE</b>		Isotropic permittivity.
<b>EPSXX</b>		Anisotropic components of permittivity.
<b>EPSYY</b>		
<b>EPSZZ</b>		
<b>EAPHASE</b>		Complex phase lag for anisotropic permittivity.

### Notes

This command defines the material characteristics for use by the analysis programs.

A set of material labels is picked using the command repeatedly, with **OPTION=PICK** and a **MATERIALLABEL** specified. A material label can be removed from the set using **OPTION=UNPICK**. If no **MATERIALLABEL** is given, the set is emptied.

Issuing the command with **OPTION=MODIFY** will modify the properties of the picked material labels to the new values given in the parameters. The value of properties associated with the material labels are unchanged if the parameter is unset.

The default value of each of the parameters is updated to be the common value of all the picked material labels. If the data of one of these parameters is unset, or the picked material labels do not share the same value, then the parameter value is left clear.

New material labels are assumed to be linear and isotropic. **OPTION=RESET** will clear the properties associated with all of the picked material labels.

The working material unit set can be changed using **OPTION=METRE**, **OPTION=CGS** or one of the other sets. If the units are changed, the values in the parameters are converted to the new unit set, and values entered will be interpreted in this unit set.

The properties of all material labels can be listed using **OPTION=LIST**.

Material labels that are not used, i.e. have no cell referencing them can be deleted using **OPTION=DELETE**. Deleting a material label that is in use will reset its properties to those of air.

All parameters can be specified. Which of these values will be used by the analysis will depend upon the analysis module being used, e.g. the phase lag properties are not used by **TOSCA**.

This command defines the **MATERIAL** properties for use by the analysis programs. All parameters can be specified for any model. Each analysis program will use the appropriate subset of properties which it needs. The values can be adjusted during creation of the database using the **SOLVER MATERIAL** sub-command.

All materials start with properties of air.

- **option=pick**: A set of material labels is selected using the command repeatedly, with **option=pick** and a material label.
- **option=modify**: After a set of material labels has been picked, the default values of each parameter will be set to the data that is common to the picked material labels. This data can be modified using **option=modify**, with the new values supplied for each of the parameters. If a value is not supplied for a parameter value, that data stored with each material label is unaffected. There are two sets of parameters, one for **anisotropy=isotropic** or **packed** and the other for **anisotropy=anisotropic**. Only isotropic properties can be set using the GUI.
- **option=reset** will reset the values of all the data of all picked material labels to the properties of air.
- **option=METRE**, **option=CGS**: The values can be specified in CGS units or SI with various length units by using Material **option=CGS**, **option=METRE** etc. If the units are changed, the values in the parameters are converted to the new unit set, and values entered will be interpreted in this unit set.
- **option=list**: The properties of all picked materials can be listed using **option=list**.
- **option=delete**: Material labels that are not used, i.e. have no volume referencing them can be deleted using **option=delete**. Deleting a material that is in use will reset its properties to those of air.

## The **MESH** Command

---

### Menu Routes:

```
MESH↓
  Surface mesh ...
  ... options
  ... triangles
  ... quadrilaterals
```

### Command Line Parameters:

Command	<b>MESH</b>	
Parameter	Default	Function
<b>NORMAL</b>	0.1	Curved surface fitting tolerance.
<b>PARAMETER</b>	<b>NO</b>	Triangulate in surface parameter space: <b>YES</b> or <b>NO</b> .
<b>UPDATE</b>	<b>YES</b>	Check that the triangulation matches the Delaunay criterion after each point is added: <b>YES</b> or <b>NO</b> .
<b>ELEMENT</b>	<b>QUADRILATERAL</b>	<b>QUADRILATERAL</b> 4-sided surface facets leading to a hexahedral volume mesh.
		<b>TRIANGLE</b> 3-sided surface facets leading to a tetrahedral volume mesh.
<b>TOL</b>	<b>1.E-6</b>	Coincident point tolerance.

The finite element mesh is created by first meshing the region surfaces using the **MESH** command. The **FILL** command ([page 4-115](#)) can then be used to generate the volume mesh.

The **MESH** command generates the surface mesh on all the pre processor surface facets from the subdivision the user has specified for the edges. The two types of elements (**TRIANGLES** and **QUADRILATERALS**) that may be selected, determine which type of volume element will be created by the **FILL** command (tetrahedra or hexahedra) ([page 4-115](#)).

If the **MESH** command fails on any surface facet because the variation of element size is too extreme or the discretisation does not obey the rules for quadrilateral meshes, that facet is given the label **DEBUG** so that it can be identified using the **DISPLAY** command (page 4-97) or the **THREED** command (page 4-171).

## Triangular meshes

Triangular meshes can be generated on any polygonal surface. The only restriction is that surfaces with more than 4 sides should be planar.

Points are automatically added to the surface to give a smooth variation of element size, given the edge subdivision. A regular triangular mesh will be created on 3 and 4 sided surfaces, if the edge subdivisions are equal on opposite sides. Additional points may be added to the surface mesh if the surface is curved. The difference between the real surface shape and the finite element approximation is checked against the **NORMAL** tolerance and points may be added to improve the finite element model.

In general the surface mesh should be created in the real coordinate system used for the model. There may be exceptional cases where the surface mesh is required to be created in the **PARAMETER** space of the surface, for example, to achieve a rapid variation in element size.

## Quadrilateral meshes

Quadrilateral meshes can only be generated if the model consists of regions which are all hexahedra, or degenerate hexahedra. The following conditions must be fulfilled in order to generate a mesh of quadrilateral surface elements:

- there are no facets with more than 4 sides
- all 4 sided facets have equal subdivisions on opposite sides
- all 3 sided facets have two sides with the same subdivision

## The **MODIFY** Command

---

### Menu Route:

MODIFY↓  
 Mesh number  
 Point coordinates  
 Subdivisions  
 Material properties  
 Boundary conditions

### Command Line Parameters:

Command	<b>MODIFY</b>	
Parameter	Default	Function
<b>MESH</b>	1	Number of finite element mesh to be modified.
<b>START</b>	1	Number of the first plane or layer. Planes are numbered: 1,2, 3, etc.; mid-extrusion planes and layers are numbered: 1.5, 2.5, 3.5, etc.
<b>THREED</b>	<b>NO</b>	Use 3d Viewer.
	<b>NO</b>	No 3d Viewer.
	<b>YES</b>	Use 3d Viewer to show the model during <b>MODIFY</b> .

The **MODIFY** command allows much of the finite element mesh data created by the **DEFINE** (page 4-49) and **EXTEND** (page 4-113) commands to be changed. The interaction is very similar to that used in **DEFINE** and **EXTEND**, and therefore only the differences will be highlighted.

Only the topology of the base plane cannot be modified. The **EDIT** command (page 4-109) can be used to change this topology.

The data of the finite element mesh is separated into 4 sections, corresponding to the modes of the **DEFINE** command. The user is asked first to choose between points, subdivisions of base plane facets and extrusion layers, materials and volume properties or boundary conditions.

<b>MODIFY Modes</b>	
Keyword	Meaning
<b>BOUNDARY</b>	Boundary conditions can be changed.
<b>MATERIALS</b>	Material names, potential types, element types and properties can be changed.

<b>MODIFY Modes (continued)</b>	
<b>POINTS</b>	Points can be moved and mesh planes transformed.
<b>QUIT</b>	Leave the <b>MODIFY</b> command.
<b>SUBDIVISIONS</b>	Subdivision of the planes and the layers can be changed. Layers can be changed between <b>LINEAR</b> and <b>QUADRATIC</b> extrusions.

Valid replies are: **BOUNDARY**, **MATERIAL**, **POINTS**, **QUIT** or **SUBDIVISIONS**.

*In menu mode* modifications can be made to one plane of layer at a time. The program prompts for a value of **PLANE NUMBER** or **LAYER NUMBER**.

*In keyboard mode*, for each section, each mesh plane or extrusion layer of the mesh specified by the **MESH** parameter, starting with the plane or layer given by the **START** parameter, is presented to the user. Mesh planes are numbered 1, 2, 3, etc., and mid-extrusion planes and layers 1.5, 2.5, 3.5, etc. Layer 1.5 is between plane 1 and plane 2.

## Point Modification Mode

### *Menu Route:*

MODIFY↓  
Point coordinates

*In menu mode*, the plane selected is presented to the user. Points can be moved individually or in groups. Individual points are selected using menu item **Move Point** and can be repositioned graphically including the use of grids and construction lines (see [page 4-53](#)) or numerically. Points can be grouped using menu item **Select/de-select point** and transformed using **Transform points**. If no points have been grouped, all the points of the plane are transformed. If mistakes are made, the point or points can be returned to their original locations using **Undo**. This must be done before leaving the **MODIFY** command.

*In keyboard mode* each plane, starting from the plane number given by the **START** parameter and ending with the top plane is presented to the user. Points can be moved individually or in groups. Individual points are selected using the **<space>** cursor hit and can be repositioned using the cursor including the use of grids and construction lines (see [page 4-53](#)) or the keyboard. Points can be grouped using **K** and transformed using **T**. If no points have been grouped, then **T** will transform all the points of the plane. If mistakes are made, the point or points can be returned to their original locations using **U** cursor hit. This must be done before finishing modifications to the plane.

Full details of the cursor commands and the transformation options are given in the following sections.

### *Point Selection Menu and Cursor Hits*

<b>Point Selection Menu and Cursor Hits</b>		
Cursor hit	Menu Item	Function
<b>&lt;space&gt;</b>	Move point	Select point nearest cursor to be moved. It can be repositioned using the Point Repositioning Mode cursor hits ( <a href="#">page 4-139</a> ).
<b>H</b>		Display menu help message explaining all the cursor options. (More help available after point selection.)
<b>K</b>	Select/de-select point	Select point nearest the cursor to be transformed. Repeating <b>K</b> for a selected point de-selects it.
<b>Q</b>	Finish Editing	Leave this plane. If on a mid-extrusion plane, move on to the top plane of the new layer. Otherwise the program asks about the next extrusion ( <i>keyboard mode</i> ) or moves on to material definitions ( <i>menu mode</i> ).

<b>Point Selection Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu Item	Function
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:
		4 numeric values <i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b> Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b> Return to previous screen size.
		<b>BOUND</b> Use bounding rectangle of geometry.
		<b>&lt;return&gt;</b> Reconstruct at the same size.
<b>T</b>	Transform points	Define general transformations for the points on the plane. (Use <b>K</b> to select points, or all points will be transformed.) The transformation options are given on <a href="#">page 4-77</a> .
<b>U</b>	Undo move or transform	Undo the last move or transform operation on this plane or layer.
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off ( <a href="#">page 4-52</a> ).

***The Point  
Repositioning  
Mode Menu and  
Cursor Hits***

<b>Point Repositioning Mode Menu and Cursor Hits</b>		
Cursor hit	Menu item	Function
<space>	At mouse	Reposition the point at the cursor cross hair position or the nearest grid point.
<b>C</b>	Give U, V, W	Switch to input from keyboard in cartesian coordinates UVW. Coordinates should be entered in free format. Default values of U, V and W are the values prior to the move. They can be accessed via the variables #1, #2 and #3. Type Q to leave point in its present position.
<b>E</b>	Remove C_line	Erase construction line closest to cursor cross hair.
<b>G</b>	Grid	Define or remove a grid.
<b>H</b>		Display menu help message explaining all the cursor options.
<b>I</b>	Enter C_lines	Input construction line specifications.
<b>L</b>	List C_lines	List construction line specifications.
<b>N</b>	On nearest C_line	Reposition the point on the nearest construction line with minimum normal distance. This only affects the U and V coordinates; W remains unchanged.
<b>P</b>	Give R, Theta, W	Switch to input from keyboard in local polar coordinates R $\theta$ W. Coordinates should be entered in free format. Default values of R, $\theta$ and W are the values prior to the move. They can be accessed via the variables #1, #2 and #3. Type Q to leave point in its present position.
<b>Q</b>	Return without moving	Leave the point at its previous position.

<b>Point Repositioning Mode Menu and Cursor Hits (continued)</b>		
Cursor hit	Menu item	Function
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:
		4 numeric values <i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b> Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b> Return to previous screen size.
		<b>BOUND</b> Use bounding rectangle of geometry.
		<b>&lt;return&gt;</b> Reconstruct at the same size.
<b>T</b>	Show coordinates	Type the coordinates of the point closest to the cross-hairs, and select it for repositioning instead of the point already selected. The coordinates are given in cartesian (UVW) and polar (R $\theta$ W) coordinates.
<b>X</b>	At C_line intersection	Reposition the point at the closest construction line intersection or end. This only affects the U and V coordinates; W remains unchanged.
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off ( <a href="#">page 4-52</a> ).

### Plane Transformation Options

The transformations can be any combination of the following:

Plane Transformation Commands	
Command	Parameters and Function
<b>CARTESIAN</b>	<i>exp_u exp_v exp_w</i> The points are moved to new positions defined by expressions for their ( <i>u</i> , <i>v</i> and <i>w</i> ) coordinates in terms of their old coordinates: <b>U</b> , <b>V</b> , <b>R</b> , <b>TH</b> and <b>W</b> . The copy number, ( <b>#COPY</b> ), can also be used.
<b>DISPLACE</b>	<i>du dv dw</i> Displace points in the current viewing local coordinate system by adding ( <i>du</i> , <i>dv</i> , <i>dw</i> ) to their coordinates ( <b>U</b> , <b>V</b> , <b>W</b> ).
<b>POLAR</b>	<i>exp_r exp_θ exp_w</i> The points are moved to new positions defined by expressions for their ( <i>r</i> , <i>θ</i> and <i>w</i> ) coordinates in terms of their old coordinates: <b>U</b> , <b>V</b> , <b>R</b> , <b>TH</b> and <b>W</b> . The copy number, ( <b>#COPY</b> ), can also be used.
<b>PROJECT</b>	<i>du dv dw ucentre vcentre wcentre uangle vangle wangle</i> Project the points in the direction ( <b>du</b> , <b>dv</b> , <b>dw</b> ) until they intersect the XY plane of a coordinate system specified by its origin ( <i>ucentre</i> , <i>vcentre</i> , <i>wcentre</i> ) and axis rotation angles ( <i>uangle</i> , <i>vangle</i> , <i>wangle</i> ).
<b>QUIT</b>	End the sequence of transformations.
<b>ROTATE</b>	<i>ucentre vcentre wcentre uangle vangle wangle</i> Rotate points by angles ( <b>uangle</b> , <b>vangle</b> , <b>wangle</b> ) around axes parallel to the local coordinate system and passing through the point ( <i>ucentre</i> , <i>vcentre</i> , <i>wcentre</i> ).
<b>SCALE</b>	<i>ucentre vcentre factor</i> Scale points by multiplying the distance from local coordinate point ( <b>ucentre</b> , <b>vcentre</b> ) by <b>factor</b> . The <b>W</b> coordinate of the points is not affected.

In *menu mode* the transformation commands are offered as menu items and the parameters are supplied using parameter boxes.

## Subdivision Modification Mode

The two mesh generators (for tetrahedra and hexahedra) have different requirements for the subdivision of mesh volumes. Hexahedral elements can only be generated in hexahedral volumes or degenerate hexahedral volumes with regular subdivision (i.e. opposite lines of each facet must have the same numbers of ele-

ments). Tetrahedral elements can be generated in any volume with any subdivision.

To support the hexahedral mesh generator, the subdivision of the facets in the mesh plane should be modified use the method described in “[Subdivision Modification Mode](#)” on page 4-141, accessed by using **START=1**. The subdivision of extrusion layers should be modified as described in “[Modifying Subdivisions of the Extrusion Layers](#)” on page 4-144, avoiding the **EDIT** option.

The **EDIT** option (“[Modifying Subdivisions for Tetrahedral Meshes](#)” on page 4-145) provides facilities for modifying the subdivision of lines in a more general way to support the flexibility of the tetrahedral mesh generator.

### *Modifying the Subdivisions of All the Planes*

```
MODIFY↓
      Subdivision → Uniform subdivision → In-plane
```

*In keyboard mode* a facet is selected for subdivision by positioning the cross-hairs just inside a facet, close to the edge to be divided and pressing the **<SPACE>** bar. The program prompts for the number of subdivisions. Subdivisions can also be set globally, i.e. all edges set to the same subdivision number, using the **G** cursor hit.

*In menu mode* the subdivision can be set first using menu item **Set subdivision** and can then be applied to a single edge or globally to all the in-plane lines in the entire mesh.

If the subdivision is given for a single edge and that edge is one side of a quadrilateral facet, then the subdivision of the opposite edge will be set to the same value. The subdivision will be carried through the mesh until a triangular facet, the edge of the mesh or a facet already set in this operation is reached. (This does not happen if a polygon has been defined on the base plane.)

When the subdivision of an edge is set, the corresponding edges in all other layers are also set to the same subdivision.

Subdivisions of edges can be changed any number of times, until the required pattern is achieved.

**Facet  
Subdivision  
Menu and  
Cursor Hits**

<b>Facet Subdivision Menu and Cursor Hits</b>		
Cursor hit	Menu item	Function
<space>	Apply to line	Select closest edge for its subdivision to be set.
<b>A</b>		Leave Facet Subdivision Modification and proceed to Extrusion Layer Modification.
<b>G</b>	Apply globally	Select all edges for their subdivisions to be set to the same value.
<b>H</b>		Display menu help message explaining all the cursor options.
<b>Q</b>		Leave Facet Subdivision Modification and proceed to Extrusion Layer Modification.
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:
		4 numeric values <i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b> Select diagonally opposite corners of the display area with <space> cursor hits.
		<b>RESTORE</b> Return to previous screen size.
		<b>BOUND</b> Use bounding rectangle of geometry.
		<return> Reconstruct at the same size.
<b>X</b>	Finish ... no checking	Leave Facet Subdivision Modification, without checking.
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.

Facet Subdivision Menu and Cursor Hits ( <i>continued</i> )		
Cursor hit	Menu item	Function
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off (page 4-52).
	Finish ... valid hex mesh?	Leave Facet Subdivision Modification, performing checks.

### *Modifying Subdivisions of the Extrusion Layers*

MODIFY↓

Subdivision → Uniform subdivision → Extrusions

The program presents a list of the current settings of the extrusion subdivisions and requests that any necessary changes are made. The list gives the layer number, the number of extrusions and the layer type. The layer type is either **LINEAR** or **QUADRATIC**; a quadratic extrusion layer has a mid-extrusion plane and the W directed lines are quadratic.

The subdivision and layer type of each layer can be changed. Changing a linear layer to a quadratic layer causes a new mid-extrusion plane of points to be added. The positions of these points is initially at the geometric mid-point of the layer, but can be changed in Point Modification Mode. Changing a quadratic extrusion to linear removes the mid-extrusion plane.

Single line commands specify the changes to be made. These consist of the layer number, the number of subdivisions and the keyword **LINEAR** or **QUADRATIC**. \* can be used for the layer number to indicate that all layers should be set the same. Three keywords are available:

- **LIST** to list the current settings
  - **EDIT PLANE** *n* to switch to variable subdivision editing on plane *n*.
  - **EDIT LAYER** *n* to switch to variable subdivision editing in layer *n*.
  - **QUIT** to leave Extrusion Layer Modification Mode.
- Example - to change layer 3 to have 5 subdivisions and be quadratic, to list the settings and then leave Extrusion Layer Modification Mode:

```
OP-SUBDIV > 3 5 q
OP-SUBDIV > list
OP-SUBDIV > quit
```

### *Modifying Subdivisions for Tetrahedral Meshes*

MODIFY↓

Subdivision → Variable subdivision

To support the flexibility of the tetrahedral mesh generator, the subdivision of the lines in mesh planes or extrusion layers can be modified using the Variable Subdivision option.

*In keyboard mode* this is selected using the **EDIT LAYER *n*** or **EDIT PLANE *n*** options in the extrusion subdivision modification mode.

The selected plane or layer is displayed so that one or more lines can be selected. Note that, for layers, the lines to be selected appear as the corners of the facets. A new number of subdivisions can be applied to the lines and also to the corresponding lines in other planes or layers using the **FROM** and **TO** keywords. The number of subdivisions can be given as an expression in terms of **N**, the existing number of subdivisions and **SIDE**, the length of the line.

### *Line Selection Menu and Cursor Hits*

Line Selection Menu and Cursor Hits		
Cursor hit	Menu item	Function
<space>	Select and define	Select line nearest cursor and prompt for subdivision.
H		Display menu help message explaining all the cursor options. (More help is available after entity selection.)
K	Select/de-select line	Select the line nearest the cursor. Repeating <b>K</b> for a selected line, de-selects it. The subdivision for all the selected lines can be given after a <SPACE> cursor hit.
Q	Finish	Leave this plane or layer.

Line Selection Menu and Cursor Hits ( <i>continued</i> )			
Cursor hit	Menu item	Function	
<b>R</b>	Re-draw picture	Reconstruct the display. The program requests a new size. Valid replies are:	
		4 numeric values	<i>umin, umax, vmin, vmax</i> Default values are the current settings.
		<b>CURSOR</b>	Select diagonally opposite corners of the display area with <b>&lt;space&gt;</b> cursor hits.
		<b>RESTORE</b>	Return to previous screen size.
		<b>BOUND</b>	Use bounding rectangle of geometry.
	<b>&lt;return&gt;</b>	Reconstruct at the same size.	
<b>V</b>	Change view	Change the UVW coordinate system. This affects the view and the extrusion direction.	
<b>Z</b>	Aspect-ratio search	Switch aspect ratio searching on or off (page 4-52).	

### *Subdivision Sub-commands in Keyboard Mode*

Following a **<space>** cursor hit, one of the following sub-commands should be given. The commands apply to the line selected with the **<space>** and to any previously selected with **K**.

The subdivision sub-commands consist of the following items:

Subdivision Sub-commands	
Command and parameter:	
<b>number</b>	Divide the line into <i>number</i> elements. <b>number</b> can be an expression in terms of <b>N</b> , the existing subdivision and <b>SIDE</b> , the length of the line.
<b>HELP</b>	Obtain help on all sub-commands and options.
<b>QUIT</b>	Abandon the currently selected lines.

<b>Subdivision Sub-commands (continued)</b>	
Optional keywords - setting multiple entities:	
<b>ALL</b>	Sets all lines in layer(s) or plane(s).
<b>FROM value</b>	Sets all layers/planes from layer/plane number given through to the current layer/plane or the layer/plane number given with <b>TO</b> .
<b>KEEP</b>	Keeps current selection of lines for another subdivision.
<b>TO value</b>	Sets all layers/planes from current layer/plane or the layer/plane number given with <b>FROM</b> through to the layer/plane number given. The value may be *to indicate the top layer/plane.

## Subdivision DialogBox in Menu Mode

Following **Select** and **define**, a value for the subdivision must be supplied in the DialogBox.

Subdivision Definition

No. of elements

Other planes or layers All lines:

From  To   All

The value can be a number or an expression in terms **N**, the existing number of subdivisions or **LENGTH**, the length of the line. Other inputs and buttons give access to other layers and to setting all lines in the layer or plane.

## Material Modification Mode

### *Menu Route:*

MODIFY↓

Material properties

The material and mesh properties within each volume from the layer specified with the **START** parameter can be changed. Each layer is presented to the user in turn and values may be set which over-ride the current settings within each volume.

The method for selecting volumes and defining materials is exactly the same as in the **DEFINE** command (page 4-78).

## Boundary Condition Modification Mode

### *Menu Route:*

MODIFY↓  
Boundary conditions

Boundary conditions can be set on any surface of any volume in the mesh. In order to achieve this the program presents the facets in 4 separate groups: first the facets on the base plane, second the planes between the first and the last, third the final plane of the mesh, and last the facets normal to the planes, one layer at a time. However, if the **START** parameter is greater than 1, then some of the planes are omitted. If **START** points to an extrusion layer then only the extrusion facets are presented.

Facets are selected and boundary conditions defined in exactly the same ways as in the **DEFINE** command ([page 4-86](#)).

## The **READ** Command

---

### Menu Routes:

**FILE**↓  
Read pre-processor file

**DEFINE**↓  
Read conductor data

### Command Line Parameters:

Command	<b>READ</b>	
Parameter	Default	Function
<b>FILE</b>	<i>none</i>	Name of OPERA-3d pre processor data file.

The **READ** command reads a pre processor data file into the program. There is one parameter which defines the name of the **FILE**. If no file name extension is given, the extension *oppre* is assumed.

OPERA-3d pre processor data files consist of all the commands issued to the program including cursor commands. Thus reading a file recreates the data in the same way as when commands are typed at the terminal except that fewer text messages appear on the screen and only the picture of the base plane is shown. All messages are written to the dialogue file *oppre.lp*.

All the commands are decoded, so that the parameter values are stored but only certain commands are executed. These are the commands which define finite element data (**DEFINE**, **EXTEND** and **MODIFY**), the conductor data (**CONDUCTOR**) and built-in commands (**\$ OS** and **\$ CD** are not executed).

If any errors are encountered while reading the file, the number of the line last read is given. This may enable correction of the file. The file has comments in it to aid the user to follow the sequence of commands. These occur in place of the cursor commands which are introduced by the character string *CURS*. Comments have the character string *\*\*\*\** followed by the number of following comment lines.

If the file does not contain all the commands necessary to complete a **DEFINE** command, normal terminal input is resumed when the end of the file is reached. It is advisable to use the **R** cursor hit first to obtain a picture of the current position within the **DEFINE** sequence.

The **READ** command should only be used to input files created using the **WRITE** command or the **WRITE** sub-command of **CONDUCTORS** or by renaming *oppre.backup*. Other command input files, such as *oppre.log* should be input using **\$ COMINPUT** (page 2-22).

Pre processor data files can be edited to contain **\$ COMINPUT** commands and a command input file can also contain a **READ** command. However, if a command input file contains a **READ** command, the file to be read must not contain a **\$ COMINPUT** command.

The **READ** command in menu mode should only be used to read complete data files, i.e. files which contain complete **DEFINE** commands. Incomplete data files can only be read in keyboard mode.

## The **REDEFINE** Command

---

### *Menu Route:*

DEFINE↓  
Redefine a mesh

### *Command line parameters:*

Command	<b>REDEFINE</b>	
Parameter	Default	Function
<b>MESH</b>	1	Number of finite element mesh to be redefined.

The **REDEFINE** command removes all the volumes from a mesh, leaving only the baseplane points, lines and facets. It then enters the same command or menu sequence as the **DEFINE** command (page 4-49) to add, remove or edit point, facets and subdivisions in the baseplane and then extrude. The only way this differs from the **DEFINE** command is that the baseplane facets from the mesh are already there when Point Definition Mode is entered (page 4-55).

This implies that all the points on the baseplane will retain their existing **u**, **v** and **w** coordinates. Any new points added will use the specified **w** coordinate.

The parameter, **MESH**, allows any mesh in a multiple mesh model to be redefined.

## The **SLIP** Command

---

### Menu Routes:

MODIFY↓  
 Add slip surface  
 Remove slip surface

### Command Line Parameters:

Command	<b>SLIP</b>	
Parameter	Default	Function
<b>OPTION</b>	<b>ADD</b>	Command option:
		<b>ADD</b> Add slip surface. <b>REMOVE</b> Remove all SLIP boundary conditions.
<b>RADIUS</b>	<b>1</b>	Radius of slip surface.
<b>RTOLERANCE</b>	<b>0.001</b>	Tolerance on <b>RADIUS</b> .

The **SLIP** command is the most convenient way to apply the **SLIP** boundary condition to the interface between the stator and rotor of CARMEN rotating machine analysis models. The slip surface should be cylindrical, extend the full length of the model, and only touch volumes modelled with **AIR** and **TOTAL** scalar potential.

- The **SLIP OPTION=ADD** command adds a slip surface by applying the **SLIP** boundary condition to all facets whose points are at the given **RADIUS±RTOLERANCE**. The label **SLIP** is also added to the points for use in the **TRANSFORM** command (page 4-178).
- The **SLIP OPTION=REMOVE** command removes the **SLIP** boundary condition from all facets which have it and also removes the label **SLIP** from their points.

## The **SOLVERS** Command

---

**Menu Route:**      **FILE**↓  
   **Analysis**

**Command Line  
Parameters:**

Command	<b>SOLVERS</b>	
Parameter	Default	Function
<b>FILE</b>	<i>none</i>	Database file name.
<b>PROGRAM</b>	<i>none</i>	Analysis program:
		<b>CARMEN</b> Rotating machines. <b>ELEKTRA</b> Low frequency eddy currents.  <b>SCALA</b> Space charge beam. <b>SOPRANO</b> High frequency. <b>TOSCA</b> Statics.
<b>TYPE</b>	<i>none</i>	Analysis type:
		<b>CURRENT</b> Current flow (TOSCA). <b>ELECTROSTATICS</b> Electrostatics (TOSCA). <b>EV</b> Eigenvalues (SOPRANO). <b>MAGNETOSTATICS</b> Magnetostatics (TOSCA).  <b>ROTATIONAL</b> Rotational velocity (ELEKTRA). <b>SSAC</b> Steady-state AC (ELEKTRA or SOPRANO).  <b>TRANSIENT</b> Transient (ELEKTRA). <b>VELOCITY</b> Linear velocity (ELEKTRA).

Command	SOLVERS (continued)		
Parameter	Default	Function	
OPTION	NEW	Database option:	
		ADD	Add a new simulation to an existing database.
		COPY	Copy a simulation within an existing database.
		DEFAULT	Equivalent to <b>NEW</b> or <b>ADD</b> depending whether there is a volume mesh or not.
		EDIT	Edit a pending simulation in an existing database.
NEW	RESTART	Create a new database.	
		Restart a completed analysis in an existing database.	
CASE	1	Simulation number to be <b>COPIED</b> , <b>EDITED</b> or <b>RESTARTED</b> .	
ELEMENT	MIXED	Element-type over-ride.	
		LINEAR	All elements linear.
		MIXED	Elements as previously set.
QUADRATIC	All elements quadratic.		
UNITS	CGS	Units of the data.	
		CGS	Practical CGS units.
		INCH	SI units with lengths in inches.
		METRE	SI units.
		MICRON	SI units with lengths in $\mu\text{m}$ .
MM	SI units with lengths in mm.		

This command creates or modifies a database file for analysis by one or more of the OPERA-3d analysis programs. The command prompts for some additional information and has a set of sub-commands which should be used to set other options and material properties. The **SOLVERS** command has been designed to be used in menu mode and the options displayed in the menu boxes are restricted to those appropriate for the type of analysis selected.

## Analysis types

The parameters **PROGRAM** and **TYPE** together select the analysis type. The following are valid combinations.

<b>OPERA-3d Analysis Types</b>			
<b>PROGRAM</b>	<b>TYPE</b>	Menu item	Meaning
<b>CARMEN</b>		Rotating Machine	Eddy currents induced by rotational motion in a transient field including non-linear materials and permanent magnets.
<b>ELEKTRA</b>	<b>SSAC</b>	Low Frequency	
		Steady-state AC	Eddy currents induced by low-frequency AC fields, including quasi-non-linear materials.
<b>ELEKTRA</b>	<b>TRANSIENT</b>	Transient	Eddy currents induced by transient fields, including non-linear materials and permanent magnets.
<b>ELEKTRA</b>	<b>VELOCITY</b>	Linear velocity	Eddy currents induced by linear motion of smooth components in a DC field, including non-linear materials and permanent magnets.
<b>ELEKTRA</b>	<b>ROTATION</b>	Rotation velocity	Eddy currents induced by rotational motion of smooth rotors in a DC field, including non-linear materials and permanent magnets.
<b>SCALA</b>		Space charge beam	Electrostatics, including particle emitters and tracking and space charge effects.

<b>OPERA-3d Analysis Types (continued)</b>			
<b>PROGRAM</b>	<b>TYPE</b>	Menu item	Meaning
<b>SOPRANO</b>	<b>SSAC</b>	High Frequency	
		Steady-state AC	High-frequency analysis data defined frequency.
<b>SOPRANO</b>	<b>EV</b>	Eigenvalues	Eigenvalue analysis of cavities.
<b>TOSCA</b>	<b>MAGNETIC</b>	Statics	
		Magnetostatics	Non-linear magnetostatics.
		Electrostatics	Non-linear electrostatics.
<b>TOSCA</b>	<b>CURRENT</b>	Current Flow	Non-linear current flow.

## Database files and file names

OPERA-3d database files are binary files. The pre processor adds a file name extension *op3*.

A database file contains one finite element mesh and one or more simulations. The finite element mesh consists of:

- the nodes
- the elements, including material names and potential types and references to pre processor facets and volume numbers
- the unit set

A simulation consists of:

- the analysis type
- the analysis options
- the boundary conditions
- the material properties
- the coils and drive specifications
- an optional title

When a database is created, the finite element data is stored in it, together with a simulation, which is initially marked as 'pending'. Additional simulations can be

added to a database. During analysis, the OPERA-3d analysis programs will solve any pending simulations of the appropriate type found in the database.

## Creating a new database

**Menu Route:** FILE↓  
 Analysis  
 ... create new database

**Command Line Parameters:** SOLVE OPTION=NEW,

Before a new database can be created, the surface and volume mesh must be generated (see “[The MESH Command](#)” on page 4-133 and “[The FILL Command](#)” on page 4-115). The program writes the database with the finite element data and creates a new simulation of the type selected.

Two sets of options are available when creating a new database: the units being used and the element type.

**Units** The OPERA-3d databases and analysis programs have one set of units. The pre processor data is converted to the internal units when the database is created. The pre processor data is interpreted in any one of five unit systems. These are:

Units					
Menu Item	CGS	SI (metres)	SI (mm)	SI (Microns)	SI (Inches)
Keyword	<b>CGS</b>	<b>SI</b>	<b>MM</b>	<b>MICRON</b>	<b>INCH</b>
length	cm	m	mm	μm	inch
magnetic flux density	gauss	tesla	tesla	tesla	tesla
magnetic field strength	oersted	amp m <sup>-1</sup>	amp m <sup>-1</sup>	amp m <sup>-1</sup>	amp m <sup>-1</sup>
magnetic scalar pot.	oersted cm	amp	amp	amp	amp
magnetic vector pot.	gauss cm	weber m <sup>-1</sup>	weber m <sup>-1</sup>	weber m <sup>-1</sup>	weber m <sup>-1</sup>
electric flux density	coulomb cm <sup>-2</sup>	coulomb m <sup>-2</sup>	coulomb m <sup>-2</sup>	coulomb m <sup>-2</sup>	coulomb m <sup>-2</sup>
electric field strength	volt cm <sup>-1</sup>	volt m <sup>-1</sup>	volt m <sup>-1</sup>	volt m <sup>-1</sup>	volt m <sup>-1</sup>
electric potential	volt	volt	volt	volt	volt
conductivity	siemen cm <sup>-1</sup>	siemen m <sup>-1</sup>	siemen mm <sup>-1</sup>	siemen μm <sup>-1</sup>	siemen inch <sup>-1</sup>
current density	amp cm <sup>-2</sup>	amp m <sup>-2</sup>	amp mm <sup>-2</sup>	amp μm <sup>-2</sup>	amp inch <sup>-2</sup>

The unit set is stored in the database and loaded into the post processor.

### **ELEMENT** *over-ride*

The element types (**LINEAR** or **QUADRATIC**) are set during the Materials Definition Mode of the **DEFINE** command (page 4-78). It is possible to over-ride these settings to set all elements to one type (**LINEAR** or **QUADRATIC**), or to leave them as previously set (**MIXED**).

Element Types	
Keyword	Meaning
<b>LINEAR</b>	All elements linear (8-noded hexahedra or 4-noded tetrahedra).
<b>MIXED</b>	Element types as set in <b>DEFINE COMMAND</b> .
<b>QUADRATIC</b>	All elements quadratic (20-noded hexahedra or 10-noded tetrahedra).

## Editing an existing database

### *Menu Route:*

```
FILE↓
  Analysis
  ... use existing database
```

### *Command Line Parameters:*

```
SOLVE OPTION=ADD,
SOLVE OPTION=COPY, CASE=n,
SOLVE OPTION=EDIT, CASE=n,
SOLVE OPTION=RESTART, CASE=n,
```

### *Adding a new simulation*

To add a new simulation to an existing database, it is necessary that the pre processor contains the corresponding model and that the surface and volume mesh are generated.

Before meshing, the **MODIFY** command (page 4-135) can be used to change the boundary conditions. Material names, potential types, point coordinates and subdivisions should not be changed. The coils can also be changed using the **CONDUCTOR** sub-command, **MODIFY** (“The **CONDUCTOR** Command” on page 4-24).

As the simulation is added, the analysis options and material properties can be set.

**NB:** Potential types which are not part of the formulation of an analysis program are treated as a valid type. TOSCA will treat **VECTOR** potential volumes as **TOTAL** scalar potential; SOPRANO will treat all volumes as **VECTOR** potential.

### *Copying a simulation*

Copying a simulation does not need the pre processor model or finite element mesh. A new simulation of the same analysis type is created. Only the material properties and analysis options can be changed.

### *Editing a simulation*

Editing a simulation does not create a new simulation. It allows the material properties and analysis options of a pending simulation to be changed.

### *Restarting a simulation*

Restarting a simulation does the same as copying, allowing the same changes, except that within the analysis program, the old solution is used as an initial solution.

## **SOLVERS Command Prompts**

The **SOLVERS** command prompts the user for some additional information:

- a title (all analysis types)
- drive information (CARMEN, ELEKTRA and SOPRANO-SS)
- eigenvalues (SOPRANO-EV)

### *Title*

Simulations can be annotated with a title. This can have as many lines of up to 80 characters as required. It is delimited by a \* as the first character of the last line. Titles can be displayed by the post processor **SHOW** command ([page 5-88](#)).

### *Drive Information*

Each conductor in an OPERA-3d model has a drive label; non-zero potential boundary conditions can also have drive labels. For steady-state ac and transient analyses, the **SOLVERS** command prompts the user for drive information for each drive label.

- **Steady-State AC:** the drive information required is the phase angle in degrees for each drive (zero corresponds to a cosine drive).

- **Transient:** the drive information specifies the shape in time of each drive using the following options:

Transient Drives		
Keyword	Meaning	Data
<b>DC</b>	$F = 1$ for all time.	none
<b>STEP</b>	$t < 0: F = 0$ $t \geq 0: F = 1$	none
<b>RAMP</b>	$t < 0: F = 0$ $t \geq 0, t \leq t_c: F = \frac{t}{t_c}$ $t > t_c: F = 1.$	$t_c$
<b>SINE</b>	$t < 0: F = F(0)$ $t \geq 0: F = \sin 2\pi ft + \phi.$	$f$ and $\phi$
<b>COSINE</b>	$t < 0: F = F(0)$ $t \geq 0: F = \cos 2\pi ft + \phi.$	$f$ and $\phi$
<b>PEAK</b>	$t < 0: F = 0$ $t \geq 0: F = a \exp(-t^2/b).$ $a$ and $b$ are chosen such that $F = 1$ at $t = t_c.$	$t_c$
<b>RISE</b>	$t < 0: F = 0$ $t \geq 0: F = 1 - \exp(-t/t_c).$	$t_c$
<b>TABLE</b> (switch on)	$t < 0: F = F(0)$ where $F(0)$ is the value of the function in the table file at time $t = 0.$ $t \geq 0: F =$ cubic splines from tabulated data in a time-table file.	<i>filename</i>
<b>TOFF</b> (table switch off)	$t < 0: F = 1$ $t \geq 0: F =$ cubic splines from tabulated data in a time-table file.	<i>filename</i>

The program prompts for the additional information needed for each drive type. In addition, the keyword **ROTATING** should be specified for any drives which are in the rotating part of a CARMEN model.

Time-table files should be created by an editor. They consist of 2 columns of numbers in free-format, the first giving values of time and the second of function value. Consecutive lines with the same value of time introduce a discontinuity of function value or time-derivative into the cubic-spline fitting. Time-table files can be displayed using the **GRAPH** command of the post processor (page 5-52).

### *Eigenvalues*

The data for the SOPRANO Eigenvalue analysis program includes the number of eigenvalues required. They can be found in a specified range of frequencies or close to a specified frequency.

### **SOLVERS** *Sub-commands*

Seven sub-commands allow material properties and analysis options to set, edited and checked.

- **MATERIAL**: to define material properties
- **SETTINGS**: to set the analysis options
- **PERIODICITY**: to define periodicity data (TOSCA only)
- **DRIVE**: to add drive fields from a table file
- **CASE**: to specify drive frequencies (ELEKTRA-SS and SOPRANO-SS), output times (ELEKTRA-TR) or output angles (CARMEN)
- **CHECK**: to check the database and display a summary of the materials and options selected
- **QUIT**: to leave the **SOLVERS** command.

## **CASE** sub-command

### *Menu Route:*

```
FILE↓
  Analysis
  ... create new database   → Drive frequencies
  ... use existing database → Output times
                           → Output angles
```

*Command Line  
Parameters:*

Sub-Command	CASE	
Parameter	Default	Function
<b>COMMAND</b>		Command options:
	<b>ADD</b>	Add a new <b>VALUE</b> to the end of the list.
	<b>DELETE</b>	Delete the item given by <b>POSITION</b> from the list.
	<b>INSERT</b>	Insert a new value after the item given by <b>POSITION</b> .
	<b>LIST</b>	List all the values.
	<b>REPLACE</b>	Replace the <b>VALUE</b> given by <b>POSITION</b> .
<b>POSITION</b>		Position in the list for <b>DELETE</b> , <b>INSERT</b> or <b>REPLACE</b> .
<b>VALUE</b>		Value for <b>ADD</b> , <b>INSERT</b> or <b>REPLACE</b> .

The **CASE** sub-command should be used to create and edit a list of multiple case data:

- ELEKTRA-SS and SOPRANO-SS: multiple frequencies. At least one frequency must be supplied.
- ELEKTRA-TR: output times. The program automatically inserts zero in the first position in the list.
- CARMEN: output angles, unless the rotation speed is zero, in which case output times must be given.

## **CHECK** sub-command

*Menu Route:*

```
FILE↓
  Analysis
    ... create new database    → Check
    ... use existing database → Check
```

**Command Line  
Parameters:**

Sub-Command	CHECK
No Parameters	

The **CHECK** sub-command should be used to ensure that all the relevant analysis options have been specified and that all materials have been given properties.

## DRIVE sub-command

**Menu Route:**

```
FILE↓
  Analysis
    ... create new database    → Add drive fields
    ... use existing database → Add drive fields
```

**Command Line  
Parameters:**

Sub-Command	DRIVE	
Parameter	Default	Function
FILE	none	Name of table file containing fields.
LABEL	none	Label for drive (not needed for TOSCA).

The **DRIVE** sub-command adds drive fields to the analysis database from a table file. The fields replace the fields from coils for a TOSCA magnetostatics analysis or form an additional drive for a CARMEN or ELEKTRA analysis.

The format of the table file is described on [page 5-18](#). The first three columns should hold the node coordinates and the second three columns the magnetic fields.

If this command is used, the adaptive RHS integrals must not be used, since only nodal values of the drive field are available (see "[SETTINGS sub-command](#)" on [page 4-167](#)).

For CARMEN and ELEKTRA analyses, the program prompts the user to provide the phase angle (steady-state ac) or the drive shape (transient) which should be used with the additional drive. The prompts and replies take the same format as for coil and boundary condition drives (section "[Drive Information](#)" on [page 4-159](#)).

## MATERIAL sub-command

### Menu Route:

```

FILE↓
  Analysis
    ... create new database   → Materials
    ... use existing database → Materials
  
```

### Command Line Parameters:

Sub-Command	MATERIAL	
Parameter	Default	Function
<b>NAME</b>	<i>none</i>	Material name. ! for list.
<b>TYPE</b>	<b>LINEAR</b>	Material type: <b>LINEAR</b> Constant properties. <b>NONLINEAR</b> Properties depend on field strength.
<b>ANISOTROPY</b>	<b>ISOTROPIC</b>	Material structure: <b>ISOTROPIC</b> Same properties in all directions. <b>PACKED</b> Laminated material (TOSCA only). <b>MULTIPLE</b> Tensor properties.
<b>CANCEL</b>	<b>NO</b>	Cancel the material command to clear defaults. <b>CANCEL</b> always reverts to <b>NO</b> after use.
<b>MUX</b>	1	Relative permeability (isotropic or $\mu_{xx}$ ).
<b>MUY</b>		Relative permeability $\mu_{yy}$ .
<b>MUZ</b>		Relative permeability $\mu_{zz}$ .
<b>MPHA</b>	0	Complex phase lag for permeability.
<b>HCX</b>	0	X component of coercive force, $H_c$ .
<b>HCY</b>	0	Y component of coercive force, $H_c$ .
<b>HCZ</b>	0	Z component of coercive force, $H_c$ .
<b>BHX</b>		BH file (isotropic or X-direction).
<b>BHY</b>		BH file (Y-direction).
<b>BHZ</b>		BH file (Z-direction).
<b>SIGX</b>	0	Conductivity (isotropic or $\sigma_{xx}$ ).
<b>SIGY</b>		Conductivity $\sigma_{yy}$ .
<b>SIGZ</b>		Conductivity $\sigma_{zz}$ .

Sub-Command	<b>MATERIAL</b> ( <i>continued</i> )	
Parameter	Default	Function
<b>SPHA</b>	0	Complex phase lag for conductivity.
<b>JEX</b>		JE file (isotropic or X-direction).
<b>JEY</b>		JE file (Y-direction).
<b>JEZ</b>		JE file (Z-direction).
<b>EPSX</b>	1	Relative permittivity (isotropic or $\epsilon_{xx}$ ).
<b>EPSY</b>		Relative permittivity $\epsilon_{yy}$ .
<b>EPSZ</b>		Relative permittivity $\epsilon_{zz}$ .
<b>EPHA</b>	0	Complex phase lag for permittivity.
<b>DEX</b>		DE file (isotropic or X-direction).
<b>DEY</b>		DE file (Y-direction).
<b>DEZ</b>		DE file (Z-direction).

When a database is created, all materials have the properties of air or the values given by the top-level **MATERIAL** command (page 4-129). The **MATERIAL** sub-command can be used as many times as necessary to edit the material properties to the ones required.

In the current version, only the BH characteristic can be non-linear; the parameters to specify JE or DE curves are ignored. See also page 4-169 for a description of how the material **TYPE** and properties are interpreted in linear and non-linear analyses.

The BH curves can be specified as **\$default** to request the default BH curve which is displayed by the first use of the **BHDATA** command (page 4-15).

If only the material name is given, then the default values for all the parameters are set as appropriate for that material. The **MATE +CANC** command clears the default values.

*In menu mode* only those properties relevant to the analysis type of the current simulation are presented for editing.

## **PERIODICITY** sub-command

### *Menu Route:*

```

FILE↓
  Analysis
  ... create new database   → Periodicity conditions
  ... use existing database → Periodicity conditions

```

### Command Line Parameters:

Sub-Command	PERIODICITY		
Parameter	Default	Function	
<b>OPTION</b>	<i>none</i>	Option:	
		<b>ADD</b>	Add a new periodicity condition.
		<b>DELETE</b>	Delete a periodicity condition.
		<b>EDIT</b>	Edit a periodicity condition.
		<b>LIST</b>	List all periodicity conditions.
<b>NUMBER</b>	<i>none</i>	Condition number for <b>DELETE</b> or <b>EDIT</b> .	
<b>XBC</b>	<i>none</i>	Translation in x-direction.	
<b>YBC</b>	<i>none</i>	Translation in y-direction.	
<b>ZBC</b>	<i>none</i>	Translation in z-direction.	
<b>TBC</b>	<i>none</i>	Euler angle $\theta$ .	
<b>PBC</b>	<i>none</i>	Euler angle $\Phi$ .	
<b>SBC</b>	<i>none</i>	Euler angle $\Psi$ .	
<b>TYPE</b>	<i>none</i>	Connection type:	
		<b>NEGATIVE</b>	Potential changes sign
		<b>POSITIVE</b>	Potential has same sign

The **PERIODICITY** sub-command connects together nodes on facets which have the **SYMMETRY** boundary condition (see [page 4-89](#)). The connection is made by use of a coordinate transformation. The points are first off-set by (**XBC**, **YBC**, **ZBC**) and are then rotated using the Euler angles (**TBC**, **PBC**, **SBC**). In most cases either the translations or the rotations will be zero. If a transformed (slave) node is coincident with an untransformed (master) node, then a periodicity condition is established between them, such that the potential of the slave is the same as (**TYPE=POSITIVE**) or the negative of (**TYPE=NEGATIVE**) the potential of the master.

Euler angles are described in detail on [page 2-31](#), but here are some examples:

Coordinate Transformation	$\theta$	$\Phi$	$\Psi$
rotation around X-axis	$\theta$	90	-90
rotation around Y-axis	$\theta$	0	0
rotation around Z-axis	0	$\Phi$	0

## QUIT sub-command

### Menu Route:

```
FILE↓
  Analysis
  ... create new database    → Return
  ... use existing database → Return
```

### Command Line Parameters:

Sub-Command	QUIT
No Parameters	

The **QUIT** sub-command closes the database and leaves the **SOLVERS** command. **QUIT** also runs **CHECK** to display the final properties of the simulation.

It is good practice to follow the **QUIT** sub-command with a **WRITE** command, so that all the **SOLVERS** sub-commands are recorded in the pre processor data file (page 4-179).

## SETTINGS sub-command

### Menu Route:

```
FILE↓
  Analysis
  ... create new database    → Linear solution
  ... use existing database → Non-linear solution
                               → Simple RHS Integrals
                               → Adaptive RHS Integrals
                               → External fields
                               → Scala iteration data
                               → Timestep options
                               → Rotation speed
                               → Use Upwinding
                               → No Upwinding
                               → Automatic potential cuts
                               → No potential cuts
```

*Command Line  
Parameters:*

Sub-Command	SETTINGS	
Parameter	Default	Function
<b>LINEAR</b>	<b>NO</b>	Linear solution: <b>YES</b> or <b>NO</b> .
<b>TYPE</b>	<b>NEWTON</b>	Iteration type (TOSCA): <b>NEWTON</b> Newton-Raphson iterations. <b>SIMPLE</b> Simple updates.
<b>NITERATIONS</b>	21	Number of non-linear iterations.
<b>TOLERANCE</b>	0.001	Non-linear convergence tolerance.
<b>RELAXATION</b>	0.1	Under-relaxation factor (SCALA only).
<b>RHS</b>	<b>ADAPTIVE</b>	RHS integral type: <b>SIMPLE</b> Trapezium rule on each element edge. <b>ADAPTIVE</b> Up to 9 gauss points on each edge.
<b>HXEXT</b>	0	X component of the external field.
<b>HYEXT</b>	0	Y component of the external field.
<b>HZEXT</b>	0	Z component of the external field.
<b>UPWINDING</b>	<b>NO</b>	Use upwinding for ELEKTRA-VL analysis: <b>YES</b> or <b>NO</b> .
<b>ITPTSTEP</b>	21	Non-linear iterations per time step (ELEKTRA-TR).
<b>UPDATE</b>	<b>SIMPLE</b>	Time stepping method (ELEKTRA-TR). <b>AD2RK</b> Adaptive 2 <sup>nd</sup> order Runge-Kutta. <b>AD4RK</b> Adaptive 4 <sup>th</sup> order Runge-Kutta. <b>F2RK</b> Fixed time step 2 <sup>nd</sup> order Runge-Kutta. <b>F4RK</b> Fixed time step 4 <sup>th</sup> order Runge-Kutta. <b>SIMPLE</b> Fixed time step Crank-Nicholson
<b>MAXADERR</b>	1	Maximum % error in adaptive time stepping.
<b>DELTAT</b>	0.001	Fixed or initial time step.

Sub-Command	<b>SETTINGS</b> ( <i>continued</i> )	
Parameter	Default	Function
<b>RPM</b>	3000	Rotation speed (CARMEN)
<b>POTENTIALCUTS</b>	<b>YES</b>	Use automatic cuts in TOSCA: <b>YES</b> or <b>NO</b> .

The **SETTINGS** command sets various analysis options.

*In menu mode* only those options relevant to the current simulation are available.

- **Linear** or **Non-linear**: Linear analyses use constant material properties; non-linear analyses update the material properties, depending on the solution and re-solve. Non-linear analyses can include some linear materials and linear analyses can include non-linear materials. In this latter case, the material property characteristic (e.g. BH curve) is used to supply the value of the material property (permeability). The slope of the curve at zero flux-density is used.
- **RHS Integrals**: The line and surface integrals of coil fields which are part of TOSCA magnetostatics, CARMEN and ELEKTRA analyses can be done using simple integrals, which only use the values of coil fields at the nodes or using adaptive integrals which use additional field points in order to reduce the errors.
- **External Fields** can be added to TOSCA magnetostatics analyses.
- **Automatic Potential cuts** can be used in TOSCA magnetostatic analysis to automatically insert potential cuts to avoid having multiply connected volumes, where a loop of total potential volume encloses a non-zero net current.
- **Periodicity** can be used in TOSCA and SCALA to avoid the need to build the full model where it is known that both field and geometry have rotational or translational symmetry.
- **SCALA Iterations** converge to a consistent set of particle trajectories and fields. The number of iterations and under-relaxation factor can be set.
- **Upwinding** is a technique to improve analysis of moving systems (ELEKTRA-VL). The analysis program reports whether upwinding is required or not.
- **Time stepping** in ELEKTRA-TR can use fixed time steps or can adjust the time step to achieve a given accuracy. If several similar models are to be analysed, the most efficient approach might be to use adaptive 4th order Runge-Kutta for one model to ascertain the appropriate time step to use. Subsequent models can then use a fixed time step method.
- **Rotation speed** in CARMEN.

## The **TABLE** Command

---

### *Menu Route:*

**FILE**↓  
Write node table file

### *Command Line Parameters:*

Command	<b>TABLE</b>	
Parameter	Default	Function
<b>FILE</b>	<i>none</i>	Name of table file.
<b>UNIT</b>	<b>CGS</b>	Name of length unit.

The **TABLE** command is provided to facilitate an interface to other programs, primarily so that coil fields from other sources can be added to an OPERA-3d database (see “**DRIVE** sub-command” on page 4-163).

The **TABLE** command can only be used after the volume mesh has been generated using the **FILL** command (page 4-115).

The coordinates are written without any unit conversion. However, the unit name (**CGS, INCH, METRE, MICRON** or **MM**) is included in the file.

The format of table files is described in “**TABLE Files**” on page 5-18.

## The **THREED** Command

---

### Menu Route:

DISPLAY↓  
3d Viewer ... refresh display

### Command Line Parameters:

Command	<b>THREED</b>		
Parameter	Default	Function	
<b>ELEMENT</b>	<b>NO</b>	Element display switch.	
		<b>NO</b>	No element sub-division.
		<b>SURFACE</b>	Subdivision on volume surfaces.
		<b>VOLUME</b>	Subdivision within volumes.
<b>MESH</b>	<b>ALL</b>	Mesh number, <b>ALL</b> or <b>NONE</b> .	
<b>TYPE</b>	<b>VOLUME</b>	Type(s) of entities to be displayed. <b>VOLUME, FACET, LINE, POINT, ALL</b> or <b>SAME</b> .	
<b>LABEL</b>	<b>NOTAIR</b>	Label(s) on entities.	
<b>L1</b>	1	First layer to be displayed.	
<b>L2</b>	*	Last layer to be displayed. * means top layer.	
<b>COILS</b>	<b>YES</b>	Conductor display switch.	
		<b>NO</b>	Conductors not displayed.
		<b>YES</b>	Conductors displayed.
<b>C1</b>	1	First conductor to be displayed.	
<b>C2</b>	*	Last conductor to be displayed. * means highest numbered conductor.	

Command	THREED (continued)		
Parameter	Default	Function	
VECTORS	CONDUCTORS	Vector display switch.	
		CONDUCTORS	Vectors show current direction on conductors.
		CURRENT	Vectors show current density direction.
		MATERIAL	Vectors show material orientation.
		NO	Vectors not displayed.
		VELOCITY	Vectors show velocity direction.

The **THREED** command starts or updates the picture in the 3d Viewer. Pictures show the three dimensional geometry of the finite element mesh and conductors using line-drawings or coloured surfaces with hidden surfaces obscured. The discretisation can be included and vectors can be drawn to indicate the vector properties of the volumes and the current directions in the conductors. Colours are used to differentiate between volumes with different material names or to show values of assigned potentials. The parameters control the parts of the model included and other options.

The orientation, position and size of the model in the 3d Viewer and the contents of the picture can be adjusted dynamically using the mouse.

## Selecting Parts of the Finite Element Model

### Menu Route:

DISPLAY↓  
3d Viewer ... select parts

The **THREED** command draws volumes and facets created with the **DEFINE** command (page 4-97). By use of the **TYPE** and **LABEL** parameters, sub-sets of the entities can be selected in order to aid visualization of the model. The parameter **TYPE** can be set to **FACET** or **VOLUME**; the parameter **LABEL** can be set to individual labels or combinations of labels. The additional label, **NOTAIR** can be used to select all material name labels except **AIR**. Abbreviated label names

can be used where the abbreviation is not ambiguous. Abbreviations can be followed by \* to indicate that all labels which match should be selected. An entity is drawn if it has any of the labels selected. For example, to display all facets with vector potential boundary conditions, **LABEL=A\* -ALL** could be used.

Labels are assigned automatically to parts of the model by the commands **DEFINE**, **EXTEND**, **MODIFY** and **SLIP**. Automatically assigned labels include **ALL**, material names, element types potential types and boundary condition types and functions. Additional labels can be given in two ways: the **LABEL** command (page 4-125) can be used to give any label to any part of the model and the **CHECK** command (page 4-18) can be used to assign the labels **DEBUG** to volumes with bad shapes and **EXTERNAL** to facets which are not shared by two volumes.

The part of the finite element mesh displayed can also be restricted by the parameters **MESH**, **L1** and **L2**. The **MESH** parameter is used to select **ALL** meshes or one particular mesh (each **DEFINE** command creates a mesh with a new number). **MESH=NONE** can be used to omit the finite element mesh completely. **L1** and **L2** are used to select a subset of the layers of the mesh. **L2** can be set to \* to indicate the top layer.

## Selecting Conductors

### Menu Route:

```

DISPLAY↓
  3d Viewer ... select parts    → Conductors
                                → Conductor numbers
  
```

The **THREED** command draws the conductors including any symmetry copies. This can be controlled by the parameters **COIL**, **C1** and **C2**. **+COIL** and **-COIL** switch the display of the conductors on and off, and **C1** and **C2** select a range of conductors for display. **C2** can be set to \* to indicate the highest numbered conductor.

## Other parameters

### *Menu Route:*

DISPLAY↓  
3d Viewer ... command style

Three different types of picture can be displayed:

- **ELEMENT=NO** gives a wire-frame view without discretisation. This is always available.
- **ELEMENT=SURFACE** gives a solid colour view showing the selected surfaces (by facet label or volume label) and the surface discretisation. This option can only be used after the **MESH** command (page 4-133) has been run successfully.

Views of facets, with **LABEL** set to a single potential name (**AX**, **AY**, **AZ**, **POTENTIAL**, **VOLTAGE** or the incident potentials) are displayed as contour plots to show the values of the potential on the facets.

- **ELEMENT=VOLUME**, **TYPE=VOLUME** gives a solid colour view of each individual element. The elements are shown shrunk in size so that internal elements are also visible. This option can only be used after the **FILL** command (page 4-115) has been run successfully.

Arrows to show the direction of the volume vector properties can be drawn inside each volume. By default, the arrows are only drawn inside the conductors to show the current direction (**VECTOR=CONDUCTORS**) but they can also show the **MATERIAL** orientation, the **CURRENT** density direction or the **VELOCITY** direction. The arrow display can be switched off completely (**VECTOR=NO**).

## The 3d Viewer Menus

The picture in the 3d Viewer can be manipulated using the mouse. The main control is the left mouse button. Moving the mouse with the left button pressed changes the view of the model. The way in which the view is changed and other options can be selected from a menu which pops up when the right mouse button is clicked. The menu options can be selected using the left mouse button.

<b>3d Viewer Menu</b>	
<b>Translate</b>	The model can be moved in any direction by dragging with the left mouse button pressed. The cursor shape is  .
<b>Rotate</b>	The model can be rotated by dragging with the left mouse button pressed. Movement up or down rotates the model around the horizontal axis. Movement from side to side rotates around the vertical axis. This is the default state when the 3d Viewer is started. The cursor shape is  .
<b>Zoom</b>	The model can be moved closer or further away by dragging with the left mouse button pressed. Movement up the window brings the model closer; movement down the window pushes the model further away. The cursor shape is  .
<b>Views</b>	The options in the <b>Views</b> sub-menu can be used to re-initialise the view or to rotate the model so it is being viewed from either the positive or negative ends of the major coordinate axes.
<b>Options</b>	The <b>Options</b> sub-menu can be used to switch on or off components of the display, e.g. the axes.
<b>Quit</b>	The <b>Quit</b> option closes the 3d Viewer. It can be restarted using the <b>THREED</b> command.

## The **TITLE** Command

---

*Menu Route:*      OPTIONS ↓  
                          Title

*Command Line  
Parameters:*

Command	<b>TITLE</b>	
Parameter	Default	Function
<b>STRING</b>	<i>none</i>	A graphics window title.
<b>POSITION</b>	TL	Graphics window title position:
		<b>BC</b>   Bottom centre.
		<b>BL</b>   Bottom left.
		<b>BR</b>   Bottom right.
		<b>TC</b>   Top centre.
		<b>TR</b>   Top right.
<b>KEEP</b>	YES	Title preservation switch:
		<b>NO</b>   Only display title once.
		<b>YES</b>   Display title on subsequent pictures as well.
<b>NOW</b>	YES	First appearance switch:
		<b>NO</b>   Display after next graphics window clear.
		<b>YES</b>   Display immediately.
<b>DATE</b>	YES	Date, time and page number switch:
		<b>NO</b>   Date, time and page number not displayed.
		<b>YES</b>   Date, time and page number displayed.
<b>EXTRA</b>	YES	Display additional model information (if available):
		<b>NO</b>   Additional model information not displayed.
		<b>YES</b>   Additional model information displayed.

The **TITLE** command controls the display of titles and other information on the graphics window. The items which can be controlled are an additional title, and the default labelling of the graphics window with date, time and page number.

- an additional title, **STRING**, can be **POSITION**ed at any one of 6 places on the graphics window. It can be displayed immediately (**+NOW**) or after the next graphics window clear (**-NOW**). It can be used once (**-KEEP**) or kept for sub-

sequent pictures (**+KEEP**). **STRING**s which contain spaces or commas or start with a non-alphabetic character must be enclosed in quotation marks ( ' ).

- the display of the **DATE**, time and page number can be switched on or off on subsequent pictures with **+DATE** and **-DATE**.
- additional information about the model (if available) can be switched on or off on subsequent pictures with **+EXTRA** and **-EXTRA**.

## The **TRANSFORM** Command

---

**Menu Route:**        **MODIFY**↓  
                              Transform labelled points

**Command Line  
 Parameters:**

Command	<b>TRANSFORM</b>	
Parameter	Default	Function
<b>LABEL</b>	<b>ALL</b>	Point label.
<b>NEWX</b>	<b>X</b>	Expression for new X-coordinates.
<b>NEWY</b>	<b>Y</b>	Expression for new Y-coordinates.
<b>NEWZ</b>	<b>Z</b>	Expression for new Z-coordinates.

The **TRANSFORM** command can be used to move a labelled set of points to new coordinates. The points must first be labelled using the **LABEL** command (page 4-125). Then the new coordinates for the points (**NEWX**, **NEWY**, **NEWZ**) can be given as values or as expressions in terms of their existing coordinates (**X**, **Y**, **Z**).

The following should be noted:

- Whenever the **LABEL** command is used to add a label to or remove a label from a volume, facet or line, the same operation is also applied to the points which define the volume, facet or line. This enables groups of points to be labelled together.
- The **SLIP** command (page 4-152) adds the label **SLIP** to all the points on the slip surface.
- The **TRANSFORM** command cannot be undone unless the inverse transformation exists.

## The **WRITE** Command

---

### Menu Route:

**FILE**↓  
Write pre-processor file

### Command Line Parameters:

Command	<b>WRITE</b>	
Parameter	Default	Function
<b>FILE</b>	<i>none</i>	Name of OPERA-3d pre processor data file.

The **WRITE** command writes an OPERA-3d pre processor data file. There is one parameter which defines the name of the **FILE**. If no file name extension is given, and extension *oppre* is assumed.

Pre processor data files consist of all the commands issued to the program including cursor commands, except that some erroneous cursor hits are omitted. Comments are included among the cursor commands to aid the user to follow the sequence of commands. Cursor commands are introduced by the character string **CURS**. Comments have the character string **\*\*\*\*** followed by the number of following comment lines.

- Example - the following is the beginning of a OPERA-3d pre processor data file:

```
DEFI
XY
0 100 0 100
****      1
  Start of point definition
CURS I   11 0.2461406E+03 0.1195938E+03 0.9931790E+02 0.3576628E+02
A 0 0 10 0 10 60
A 0 0 55 0 55 60
A 0 0 100 0 100 60
L 0 0 100 0 0
L 0 0 100 0 60
Q
CURS X   15 0.2475313E+03 0.5979688E+02 0.1001073E+03 0.1821565E+01
CURS X   15 0.1654844E+03 0.2058125E+03 0.5353201E+02 0.8470982E+02
CURS X   15 0.1293281E+03 0.1348906E+03 0.3300729E+02 0.4444981E+02
CURS X   15 0.8343750E+02 0.7787500E+02 0.6956694E+01 0.1208392E+02
CURS X   15 0.6953125E+02 0.5423438E+02-0.9374268E+00-0.1336082E+01
CURS X   15 0.1001250E+03 0.5423438E+02 0.1642964E+02-0.1336082E+01
CURS X   15 0.1682656E+03 0.5423438E+02 0.5511083E+02-0.1336082E+01
CURS I   11 0.1682656E+03 0.5423438E+02 0.5511083E+02-0.1336082E+01
L 0 0 10 0 30
L 0 0 100 0 30
Q
```

```

CURS X 15 0.8900000E+02 0.6675000E+02 0.1011434E+02 0.5768625E+01
CURS X 15 0.1724375E+03 0.9734375E+02 0.5747906E+02 0.2313569E+02
CURS X 15 0.2308438E+03 0.1279375E+03 0.9063436E+02 0.4050275E+02
CURS Q 6 0.2308438E+03 0.1279375E+03 0.9063436E+02 0.4050275E+02
****
1
Start of face definition
CURS C 9 0.2308438E+03 0.5006250E+02 0.9063436E+02-0.3704319E+01
CURS M 10 0.2308438E+03 0.1251563E+03 0.9063436E+02 0.3892393E+02
CURS C 9 0.1613125E+03 0.1835625E+03 0.5116376E+02 0.7207923E+02
CURS C 9 0.1265469E+03 0.1585313E+03 0.3142846E+02 0.5786982E+02
CURS M 10 0.1501875E+03 0.1029063E+03 0.4484847E+02 0.2629334E+02
CURS C 9 0.1613125E+03 0.5006250E+02 0.5116376E+02-0.3704319E+01
CURS F 11 0.1515781E+03 0.5006250E+02 0.4563788E+02-0.3704319E+01
****
1
End of face
CURS C 9 0.1515781E+03 0.5006250E+02 0.4563788E+02-0.3704319E+01
CURS M 10 0.1515781E+03 0.1112500E+03 0.4563788E+02 0.3102981E+02
CURS C 9 0.1223750E+03 0.1404531E+03 0.2906023E+02 0.4760746E+02
CURS C 9 0.7648438E+02 0.7787500E+02 0.3009633E+01 0.1208392E+02
CURS M 10 0.8621875E+02 0.6814063E+02 0.8535518E+01 0.6558037E+01
CURS C 9 0.8621875E+02 0.5284375E+02 0.8535518E+01-0.2125494E+01
CURS F 11 0.8621875E+02 0.5284375E+02 0.8535518E+01-0.2125494E+01
****
1
End of face
CURS C 9 0.8621875E+02 0.5284375E+02 0.8535518E+01-0.2125494E+01
CURS M 10 0.8621875E+02 0.6396875E+02 0.8535518E+01 0.4189801E+01
CURS C 9 0.7370313E+02 0.7092188E+02 0.1430809E+01 0.8136862E+01
CURS C 9 0.7370313E+02 0.5423438E+02 0.1430809E+01-0.1336082E+01
CURS F 11 0.7370313E+02 0.5423438E+02 0.1430809E+01-0.1336082E+01
****
1
End of face

```



# **OPERA-3D REFERENCE MANUAL PART 2**

Vector Fields Limited  
24 Bankside  
Kidlington  
Oxford OX5 1JE  
England

Copyright © 1999-2002 by Vector Fields Limited, England

This document was prepared using Adobe® FrameMaker®.

UNIX is a trademark of X/Open Company Ltd.

HPGL is a trademark of Hewlett-Packard Incorporated

PostScript is a trademark of Adobe Systems Incorporated

Windows 95/98/NT is a trademark of Microsoft Corporation

X Window System is a trademark of Massachusetts Institute of Technology

I-DEAS is a trademark of Structural Dynamics Research Corporation

OpenGL is a trademark of Silicon Graphics Corporation

All other brand or product names are trademarks or registered trademarks  
of their respective companies or organisations.

# Chapter 5

## OPERA-3d Post Processor

### Introduction

---

---

The OPERA-3d post processor displays and performs further calculations on results from electromagnetic field analysis programs including CARMEN, ELEKTRA, SCALA, SOPRANO and TOSCA. The analysis programs use finite elements to model three dimensional electromagnetic devices. The post processor provides facilities to view the finite element data, with superimposed contours of results and to process and display the results calculated along lines or on two dimensional areas, in addition to specific functions such as body forces and trajectory calculations.

The OPERA-3d post processor can be used interactively on PCs, UNIX workstations, or X-terminals having GLX support.

The program is used to display three dimensional finite element models from direct access database files created by the analysis programs. Alternatively, conductor only problems can be evaluated by entering the conductor data into the post processor directly using the **CONDUCTOR** command.

Many results databases can be available to the program (activated) at any one time. It is possible for the program to look for a field point in all the active files until it is found or to restrict its searching to one resident (loaded) file. This enables the calculation of particle trajectories which extend through a system of magnets which have been analysed separately. When activating a database a local coordinate system can be specified to enable the individual magnets to be orientated correctly with respect to each other. Reflection codes and symmetries can also be given to replicate the finite element mesh so that the complete model is available to OPERA-3d even if only a small section of it was analysed and the rest was implied by boundary conditions.

When the OPERA-3d post processor is started or restarted with the **CLEAR** command, the program looks for a file called *opera.comi* in the current file directory. If such a file exists it is read into the program as a **\$ COMINPUT** file (see section “**Command Input Files**” on page 2-22). This allows the user to reset the default values of certain commands, e.g. **COLOUR** and **UNITS** or define frequently used **CONSTANTS** and **PARAMETERS** each time the program is started.

Two methods of command and data entry are available:–

1. Menu system or GUI (Graphical User Interface) - command selection and data specification are carried out under mouse control. Refer to the User Guide for information on the GUI.
2. Command line input - command selection and data specification are carried out from the keyboard.

## Post Processor Quick Reference Guide

---

The following is a complete list of the commands which can be entered in response to the console prompt 'OPERA-3d >'. Following sections contain complete descriptions of all the commands and sub-commands of the OPERA-3d post processor.

### Commands for Keyboard Entry – Command Line

Command line entry is carried out in the console window.

- Commands to make database files available to OPERA-3d:

<b>ACTIVATE</b>	Activate a result database file, specifying reflections, symmetries and local coordinate system.
<b>LOAD</b>	Make an active database file resident.
<b>SHOW</b>	List the names and some details of the active database files.
<b>SYSVARIABLE</b>	Add, delete or list system variables (solution vectors).
<b>BHDATA</b>	List the BH tables used in the loaded file.

- Conductor definition and editing command:

<b>CONDUCTOR</b>	Define, modify, erase, list, import and export conductor data.
------------------	--

- Units selection command:

<b>UNITS</b>	Select units for display and evaluation
--------------	---

- Display commands:

<b>SELECT</b>	Select objects to be displayed.
<b>THREED</b>	Displays <b>SELECTION</b> of mesh and conductors with surface contours.
<b>WINDOW</b>	Show or hide parts of the display.

- Field evaluation and display commands:

<b>SET</b>	Set field calculation options and local coordinate system for definition of field points.
<b>POINT</b>	Calculation of functions of the field at a point.
<b>LINE</b>	Calculation of functions of the field along a line.
<b>ARC</b>	Calculation of functions of the field along a circular arc defined by end points and centre.

<b>CIRCLE</b>	Calculation of functions of the field along a circular arc defined by radius, angles and local coordinate system.
<b>PLOT</b>	Display the results from the <b>ARC</b> , <b>CIRCLE</b> or <b>LINE</b> commands. <b>PLOT</b> also calculates the line integral.
<b>FIT</b>	Fit Fourier series to <b>ARC</b> , <b>CIRCLE</b> or <b>LINE</b> results or evaluate field fields on a sphere and fit Legendre polynomials.
<b>CARTESIAN</b>	Evaluate fields over a quadrilateral patch defined in cartesian (XYZ) coordinates.
<b>POLAR</b>	Evaluate fields over a quadrilateral patch defined in cylindrical polar (R $\theta$ Z) coordinates.
<b>MAP</b>	Display a contour maps or histograms of fields or field vectors evaluated with <b>CARTESIAN</b> or <b>POLAR</b> commands. <b>MAP</b> also calculates the surface integral.

- Field integration commands:

<b>INTEGRATE</b>	Integrate Maxwell Stress on <b>SELECTED</b> surface to give forces.
<b>SURFACE</b>	Integrate any field component on the <b>SELECTED</b> surface.
<b>BODY</b>	Integrate body forces and torques on conductors: $\mathbf{J} \times \mathbf{B}$ .
<b>ENERGY</b>	Calculate the stored energy, Lorentz force and power loss.
<b>VOLUME</b>	Integrate any field component over the volume of a material.

- Particle tracking commands:

<b>TRACK</b>	Calculate charged particle trajectories.
<b>COMBINE</b>	Combine magnetostatic and electrostatic fields for particle tracking.
<b>VIEW</b>	Display particle trajectory intercepts.

- Interface and utility commands:

<b>GRID</b>	Evaluate fields over a 3D grid and put answers in a file.
<b>ARITHMETIC</b>	Combine values from two table files.
<b>TABLE</b>	Input tables of coordinates, etc., and output tables of coordinates, fields, etc.
<b>IDEAS</b>	Write or append mesh and results to an I-DEAS Universal File
<b>GRAPH</b>	Draw graphs of data in text files.

- Program management commands:

<b>COLOUR</b>	Enquire and set colours for the display.
<b>TITLE</b>	Control screen titles.
<b>CLEAR</b>	Clear program data and re-initialize all commands.
<b>PICTURE</b>	Copy the picture to the clipboard or a bitmap file.
<b>PRINT</b>	Send the picture to a printer.
<b>MOUSE</b>	Swap functionality of middle and right mouse buttons.

- Ending the program:

<b>END</b>	End OPERA-3d.
------------	---------------

## Menu System – GUI

The following is an overview of the first level menu structure. A brief description of the functionality of each item is also given.

File	View	Options	Fields	Integrals	Trajectories	Tables	Conductors	3d view	Windows	Help
------	------	---------	--------	-----------	--------------	--------	------------	---------	---------	------

### File

Open	activate and load an OPERA-3d database	<a href="#">page 5-21</a>
Reload active database	re-load in a previously loaded file	<a href="#">page 5-67</a>
List active databases	list details of active databases	<a href="#">page 5-88</a>
Close loaded database	closes the currently loaded database	<a href="#">page 5-67</a>
Graph BH data	plot graphs of BH curves used in analysis	<a href="#">page 5-27</a>
Print	sends the picture to a printer	<a href="#">page 5-79</a>
Copy to clipboard	copies the picture to the clipboard or	<a href="#">page 5-72</a>
Copy to file	a bitmap file	
Commands in	read command input file	<a href="#">page 2-22</a>
Change directory	change the current directory	<a href="#">page 2-27</a>
Graph data in text file	plot graph of external data	<a href="#">page 5-52</a>
Clear all data	operating system commands	<a href="#">page 2-27</a>
Recent files	activate and load recently used data-bases	<a href="#">page 5-21</a>
Exit	end post processing session	<a href="#">page 5-45</a>

### View

Select	select parts of model to be displayed	<a href="#">page 5-80</a>
Default select and refresh	select according to simulation type and refresh display	<a href="#">page 5-80</a>
Repeat select and refresh	select without changing list of labels and refresh display	<a href="#">page 5-80</a>
3d display	control the style and view of the display	<a href="#">page 5-101</a>
Refresh	refresh the display	<a href="#">page 5-101</a>
Parts of the display	show or hide parts of the display	<a href="#">page 5-119</a>
Views	refresh with predefined views	<a href="#">page 5-101</a>
Change colours	change the colours used for the display	<a href="#">page 5-35</a>
Title	set a title for the display	<a href="#">page 5-100</a>

### Options

User variables	set and list user defined parameter and constants	<a href="#">page 2-19</a>
Field calculations	options which affect the way fields are calculated, the locations of field points and the angle around the ac cycle	<a href="#">page 5-85</a>
Field points		
AC time		
Add system variables	make available or delete additional system variables and list the available variables	<a href="#">page 5-92</a>
Delete system variables		
List system variables in database		
List system variables in program		
Include Jc in coils	include current density in coils as a system variable	<a href="#">page 5-87</a>
Units	set the units to be used for all physical quantities	<a href="#">page 5-107</a>
Toggle right mouse button	swap use of middle and right buttons	<a href="#">page 5-71</a>

### Fields

Fields at a point	evaluate field at 3d coordinate	<a href="#">page 5-76</a>
Fields on an arc	evaluate field along an arc defined by ends and centre	<a href="#">page 5-24</a>
Fields on a circle	evaluate field along an arc defined by centre and radius	<a href="#">page 5-33</a>
Fields on a straight line	evaluate field along a straight line	<a href="#">page 5-65</a>
Plot graph of field values	display fields as a graph	<a href="#">page 5-73</a>
Fit Fourier series to values	find Fourier coefficients to match values calculated	<a href="#">page 5-50</a>
Fields on a cartesian patch	evaluate fields over a 2d surface defined in xyz	<a href="#">page 5-31</a>
Fields on a polar patch	evaluate fields over a 2d surface defined in r, $\theta$ ,z	<a href="#">page 5-77</a>
Contour or vector map	display fields as a contour, vector map, or histogram	<a href="#">page 5-68</a>
Fit Legendre polynomials to values	evaluate fields on the surface of a sphere and fit Legendre polynomials to the values	<a href="#">page 5-50</a>
Fields on the surface of the model	display field contours on the surface of the model	<a href="#">page 5-101</a>

### Integrals

Energy, power and force	calculate stored energy, power loss and Lorentz forces by volume integration	<a href="#">page 5-46</a>
Other volume integrals	integrate any quantity over volumes	<a href="#">page 5-116</a>
Maxwell stress on selected surface	integrate Maxwell stress on selected surface	<a href="#">page 5-62</a>
Other surface integrals	integrate an quantity on selected surface	<a href="#">page 5-90</a>
Lorentz forces in conductors	calculate Lorentz forces in conductors	<a href="#">page 5-28</a>

### Trajectories

Calculate particle trajectories	calculate the trajectory of particles through the magnetic or electric fields	<a href="#">page 5-104</a>
Display Particle trajectories	display and process calculated trajectories	<a href="#">page 5-110</a>
Graph trajectories		
Intersect trajectories with patch		
Current density map		
Create flux tubes		
Use magnetic and electric fields	use both magnetic and electric fields when calculating trajectories	<a href="#">page 5-37</a>

### Tables

Read and write table files	create or process an OPERA-3d table file	<a href="#">page 5-97</a>
Table of field values on a grid	calculate field values at a 3-dimensional grid of points and store them in a table file	<a href="#">page 5-55</a>
Arithmetic on tables files	combine the values in 2 table files, creating a third	<a href="#">page 5-26</a>
SDRC I-DEAS Universal file	create or append to a Universal file	<a href="#">page 5-57</a>

### Conductors

New solenoid	define a new conductor	<a href="#">page 5-39</a>
New racetrack		
New bedstead		
New helical end		
New constant perimeter end		
New straight bar		
New arc		
New 8-node brick		
New 20-node brick		

Toggle conductor picking	allow graphical selection of conductors	page 5-39
Select conductors	form a list of conductors for modify, list, erase, export or body forces	page 5-39
Modify selected conductors	modify list of conductors	page 5-39
Erase selected conductors	erase list of conductors	page 5-39
List conductor data	list data of conductors in list or all conductors if list is empty	page 5-39
Import conductors from file	read a file of conductor data	page 5-39
Export conductors to file	store in a file conductors in list of conductors if list is empty	page 5-39

### Windows

Close window	close the window
Cascade	arrange 2d and 3d windows in a cascade
Tile	arrange 2d and 3d windows side by side.
2d graphics	window for graphs
3d graphics	window for 3d views of the model
Dock/Undock console	make the console part of the post processor frame or as a separate window
View/Hide console	make the console visible or not
View/Hide information box	make the information box visible or not
Save window settings	save window size and options

### Help

Contents	show contents of help files (Reference Manual)
About	show version number and support information

## Field Component Evaluation

---

The command decoder used with OPERA-3d is described fully in chapter 2. More details are given here concerning the way in which the post processor makes use of the expression analyser to evaluate output field components.

Several commands, **THREED**, **FIT**, **MAP**, **POINT**, **PLOT**, etc., have parameters **COMPONENT**, **VX**, **VY**, and **VZ** to define scalar and vector field components to be displayed. Expressions can be used to define these output field quantities with the variables being System Variables, user constants, user parameters and the parameters of the commands.

The expression for **COMPONENT**, **VX**, **VY** and **VZ** used with any command becomes the default value for all the commands which use those parameters. The initial value for **COMPONENT** is **X** and **VX**, **VY** and **VZ** are initially set to **X**, **Y** and **Z**. (These are used as defaults because they are always available.)

The **\$ CONSTANT** and **\$ PARAMETER** commands (page 2-19) can be used to perform further calculations on the results of the commands.

### System Variables

System Variables hold the data which can be used for point values, line graphs, contours and further calculations in the OPERA-3d post processor. Some system variables, such as field point coordinates, are available for all analysis types but others only have values if they can be read from an analysis database. System variables of this second type are the results of analysis. For example, a TOSCA magnetostatics analysis returns the magnetic scalar potential, field strength, flux density and source field strength (a scalar and three vectors).

Most loading of system variables happens automatically within the program, variables being loaded when they are required. For example, for TOSCA magnetostatics, the potential, field strength and flux density are loaded when a database is activated and loaded. The source field strength is loaded automatically if integral coil fields are selected in the **SET** command (page 5-85).

However, the user might sometimes need a variable which is available in the database but has not yet been loaded, or, for reasons of efficiency, might want to delete some of the system variables which have been loaded. The **SYSVARIABLE** command (page 5-92) provides this functionality as well as listing the variables in the database and program. The calculation of the values of the source current

density vector for field points inside conductors is selected using the **SET** command (page 5-85).

In post processing results from ELEKTRA-SS or SOPRANO-SS, the system variables represent the real and imaginary parts of complex quantities and additional expressions are defined to give the instantaneous values. For other types of analysis only the real parts are available.

When system variables are loaded the program follows this procedure:

- **Scalars**, for example the magnetic scalar potential, **POT**: The program reads the real part (**RPOT**). If the imaginary part (**IPOT**) is also available it will be read as well. The following expressions will be defined:

- Real scalars:

```
POT=RPOT
```

- Complex scalars:

```
POT=RPOT*COST+IPOT*SINT
POT0=SQRT (RPOT**2+IPOT**2)
POTP=ATAN2 ( IPOT ; RPOT)
```

where **COST** and **SINT** are calculated from the value of AC time defined by the **SET** command (page 5-85).

The following variables can be used by the user: **POT**, **RPOT** and (for steady state ac analyses) **IPOT**, **POT0** and **POTP**.

- **Vectors**, for example the magnetic field strength, **H**: The program reads each component (real and imaginary parts if available) and defines the following expressions.

- Real vectors:

```
HX=RHX
```

- Complex vectors:

```
HX=RHX*COST+IHX*SINT
HX0=SQRT (RHX**2+IHX**2)
HXP=ATAN2 ( IHX ; RHX)
```

and similarly for the Y and Z components. It also defines

```
HMOD=SQRT (HX**2+HY**2+HZ**2)
```

to give the magnitude of the field strength.

The following variables can be used by the user: **HX**, **RHX**, **HY**, **RHY**, **HZ**, **RHZ**, **HMOD**, and (for steady-state ac analyses) **IHX**, **IHY**, **IHZ**, **HX0**, **HY0**, **HZ0**, **HXP**, **HYP** and **HZP**.

- **Edge or Face Vectors,  $B_*$  and  $E_*$** : The program interpolates the edge or face values to provide vector quantities, e.g.  $E_X$ ,  $E_Y$  and  $E_Z$  and  $E_{MOD}$ , at each field point.

The following variables can be used by the user:  $E_X$ ,  $RE_X$ ,  $E_Y$ ,  $RE_Y$ ,  $E_Z$ ,  $RE_Z$ ,  $E_{MOD}$ .

In the lists of variables below, only the stem names are given (e.g.  $H$ ); it should be understood that all the components are also implied.

When system variables are removed, they are marked as no longer being available within the expression analyser software.

**Unit conversion** is performed on all system variables.

### *System Variables Defined in the Software*

Field point geometry	
$X, Y, Z$	Field point coordinates
$TX, TY, TZ$	Tangential unit vector to lines
$NX, NY, NZ$	Normal unit vector to surfaces

Source Quantities		
$JC$	Source current density	vector
$HC$	Coercive field	vector
$VEL$	Velocity	vector
$CHARGE$	Charge density	scalar

### *System Variables in Analysis Databases*

System Variables in Analysis Databases		
CARMEN		
$POT$	Magnetic scalar potential	scalar
$V$	Electric scalar potential	scalar
$A$	Magnetic vector potential	vector
$H$	Magnetic field strength	vector
$HS$	Source magnetic field strength	vector
$B$	Magnetic flux density	vector
$E$	Electric field strength	vector
$J$	Eddy current density	vector

<b>System Variables in Analysis Databases (continued)</b>		
<b>ELEKTRA</b>		
<b>POT</b>	Magnetic scalar potential	scalar
<b>V</b>	Electric scalar potential	scalar
<b>A</b>	Magnetic vector potential	vector
<b>H</b>	Magnetic field strength	vector
<b>HS</b>	Source magnetic field strength	vector
<b>B</b>	Magnetic flux density	vector
<b>E</b>	Electric field strength	vector
<b>J</b>	Eddy current density	vector
<b>ERRB</b>	Error in flux density	vector
<b>SCALA and TOSCA electrostatics</b>		
<b>V</b>	Electric scalar potential	scalar
<b>E</b>	Electric field strength	vector
<b>D</b>	Electric flux density	vector
<b>ERRD</b>	Error in flux density	vector
<b>SOPRANO-EV</b>		
<b>E_</b>	Edge values of electric field strength	vector
<b>B</b>	Nodal values of magnetic flux density	vector
<b>D</b>	Electric flux density	vector
<b>E</b>	Nodal values of electric field strength	vector
<b>H</b>	Magnetic field strength	vector
<b>SOPRANO-SS</b>		
<b>E_</b>	Edge values of electric field strength	vector
<b>B</b>	Magnetic flux density	vector
<b>D</b>	Electric flux density	vector
<b>E</b>	Electric field strength	vector
<b>H</b>	Magnetic field strength	vector
<b>J</b>	Current density	vector
<b>TOSCA current flow</b>		
<b>V</b>	Electric scalar potential	scalar
<b>E</b>	Electric field strength	vector
<b>J</b>	Current density	vector
<b>ERRJ</b>	Error in current density	vector

System Variables in Analysis Databases ( <i>continued</i> )		
TOSCA magnetostatics		
<b>POT</b>	Magnetic scalar potential	scalar
<b>H</b>	Magnetic field strength	vector
<b>HS</b>	Source magnetic field strength	vector
<b>B</b>	Magnetic flux density	vector
<b>ERRB</b>	Error in flux density	vector

In TOSCA current flow, the magnetic field strength, **H** is also available if integral fields are selected (“[The SET Command](#)” on [page 5-85](#))

### *Material properties at field point*

Material properties, permeability, permittivity and conductivity can only be calculated from the appropriate field quantities. For example, the isotropic permeability is given by

BMOD/HMOD

### *Solution values*

These are set when eigenvalue, steady-state ac or transient results are activated.

Solution values	
<b>FREQ</b>	Frequency
<b>TTIME</b>	Transient time

The variable **CASES** holds the number of simulations in the active database.

### *Local coordinate system*

The field vectors are defined with respect to the Global Coordinate System. The rotation matrix defined by the Euler angles of the **SET** command ([page 5-85](#)) can also be used in the expressions. The local X component of the flux density is given by

COMP=BX\*ROTL11+BY\*ROTL21+BZ\*ROTL31, etc.

Local coordinate system	
<b>ROTL11, ROTL12, ..., ROTL33</b>	Rotation matrix to convert global field vectors to local coordinate system.

## Results of commands

After the commands **BODY**, **ENERGY**, **INTEGRATE**, **MAP**, **PLOT** and **THREED** system variables are updated with the results of the calculations.

Results of commands	
<b>MINIMUM, MAXIMUM</b>	Extreme values
<b>FX, FY, FZ</b>	Forces
<b>FMOD</b>	Magnitude of force: $\text{SQRT}(\text{FX}^{**2}+\text{FY}^{**2}+\text{FZ}^{**2})$
<b>TORQX, TORQY, TORQZ</b>	Torques
<b>TMOD</b>	Magnitude of torque: $\text{SQRT}(\text{TORQX}^{**2}+\text{TORQY}^{**2}+\text{TORQZ}^{**2})$
<b>ENERGY, POWER, VOLUME</b>	<b>ENERGY</b> command integrals
<b>INTEGRAL</b>	Line and surface integrals
<b>J</b>	Charged beam current density
<b>PJ</b>	Charged beam power density

## Constants

Constants	
<b>PI</b>	$\pi$
<b>MU0</b>	$\mu_0$ , permeability of free space (SI)
<b>EPSILON0</b>	$\epsilon_0$ , permittivity of free space (SI)
<b>C</b>	$c$ , speed of light (SI)

## User constants

User constants are defined and examined using the **\$ CONSTANT** command. They allow the current value of system variables or expressions to be stored for use in subsequent calculations (see [page 2-19](#))

## User parameters

User parameters are defined and examined using the **\$ PARAMETER** command. They allow expressions to be stored. The value of user parameters is recalculated from the expression each time it is referenced using the current values of any other parameters or variables (see [page 2-19](#))

## Expressions

Expressions cannot exceed 75 characters, since they cannot be continued on subsequent lines. Full details of the operators and functions allowed in expressions are described on [page 2-8](#). The **\$ PARAMETER** command should be used to 'program' OPERA-3d if more complicated expressions are needed.

## Examples

Two examples are given here. The first example gives expressions to calculate the Maxwell Stress on a surface using the integral given by the **MAP** command. The force densities are given by **#FX**, **#FY** and **#FZ**. The parameters **#X0**, **#Y0** and **#Z0** should be previously set to the action point for the torques, **#TQX**, **#TQY** and **#TQZ**. The expressions given assume that the units are set to SI (see "The **UNITS Command**" on [page 5-107](#)).

- Example - to calculate forces and torques (output from the program is not shown):

```
$ para #hn hx*nx+hy*ny+hz*nz
$ para #hmod bx*bx+by*by+bz*bz
$ para #fx bx*#hn-nx*#hmod/2
$ para #fy by*#hn-ny*#hmod/2
$ para #fz bz*#hn-nz*#hmod/2
$ para #tqx (y-#y0)*#fz-(z-#z0)*#fy
$ para #tqy (z-#z0)*#fx-(x-#x0)*#fz
$ para #tqz (x-#x0)*#fy-(y-#y0)*#fx
```

The above commands are only valid if the field points used are all in air. If the integration surface passes through ferromagnetic material, the following set of commands should be used instead. These calculate the forces assuming an infinitely thin gap at the integration surface.

```
$ para #han (nx*bx+ny*by+nz*bz)/mu0
$ para #hanx nx*#han
$ para #hany ny*#han
$ para #hanz nz*#han
$ para #hax hx-nx*(nx*bx+ny*by+nz*bz)+#hanx
$ para #hay hy-ny*(nx*bx+ny*by+nz*bz)+#hany
$ para #haz hz-nz*(nx*bx+ny*by+nz*bz)+#hanz
$ para #hn #hax*nx+#hay*ny+#haz*nz
$ para #hm2 #hax*#hax+#hay*#hay+#haz*#haz
$ para #fx (2*#hax*#hn-nx*#hm2)*mu0/2.0
$ para #fy (2*#hay*#hn-ny*#hm2)*mu0/2.0
$ para #fz (2*#haz*#hn-nz*#hm2)*mu0/2.0
```

It is sometimes necessary to store the current value of a system variable or expression. The **\$ CONSTANT** command copies the current value of a variable or expression into a user defined name. Thus to plot the homogeneity of the flux density along a line the following set of commands could be used. First, the field at the reference point (10,0,0) is calculated and stored in a constant.

- Example - to calculate the homogeneity of flux density

```
poin 10 0 0 by  
$ cons #byrf by
```

The value of **BY** assigned to the constant **#BYRF** corresponds to value at the point (10, 0, 0). Next the homogeneity can be calculated.

```
line 0 0 0 20 0 0 100  
plot comp=(by-#byrf)/#byrf
```

## Post Processor Data Files

---

The OPERA-3d post processor reads data from the direct access database files created by the Modeller or pre processor and updated by the analysis programs. Other files can be created and read by the post processor for use by the results display commands.

### OPERA-3d Database Files

OPERA-3d database files contain all the information used and calculated by the analysis programs. This includes the finite element mesh including boundary conditions, the material properties and BH curves, the conductors and solution options. These files can only be read by the OPERA-3d. The **SHOW** command can be used to list the contents of database (page 5-88).

### GRID files

The **GRID** command writes a limited number of field values to a file to enable users to interface to their own post processing software. The files can be text or binary.

The following items are output: **X, Y, Z** and up to 9 other items which can be specified by the user.

In *text* files the table file format is used (see page 5-18).

*Binary* files contain one record per field point with up to 12 DOUBLE PRECISION (8 byte) values.

### TABLE Files

Table file format is used by the following commands:

<b>ARITHMETIC</b>	reads values from two table files and combines them, writing the answers to a third table file.
<b>FIT</b>	reads coordinates and field values from a file and fits a Fourier series.
<b>GRID</b>	writes coordinates and field values at a grid of points to a table file.

<b>MAP</b>	reads coordinates and field values from a table file and draws a contour or vector map.
<b>PLOT</b>	reads coordinates and field values from a table file and plots a graph.
<b>PRINT</b>	writes coordinates at field values of points in the display buffer to a table file.
<b>TABLE</b>	read and writes table files.

Table files are text files with some formatted and some free-format data.

**Record type 1:** The first line of a table file contains the number of data points in the file. This is given as 3 numbers, so that 1, 2 or three dimensional arrays of points can be stored. For example, a total of 1000 points could be stored as a line with record 1 containing

```
1000 1 1
```

or as a 2 dimensional patch, with 40 points on side 1 and 25 points on side 2, indicated by

```
40 25 1
```

or as a 3 dimensional grid of points:

```
10 10 10
```

The record is free-format.

**Record type 2:** The second record is repeated up to 12 times, once for each column in the file. The record holds an integer (the column number), followed by the column description of up to 80 characters. In most contexts, the string is compared to X, Y or Z to identify the coordinates, but otherwise is not decoded.

The equivalent `FORTTRAN` format is `(I2, 1X, A)`.

**Record type 3:** The third record has the same format as record 2 but the integer value must be zero. It indicates the end of the list of column headings and holds in the character string the unit name associated with the file. The units names which are recognised are: `[CGS]`, `[METRE]`, `[MM]`, `[MICRON]` and `[INCH]`.

**Record type 4:** The final record contains the data and is repeated for as many points as indicated by the product of the numbers in record 1. There should be up to 12 values on each line, as many as the column headings give by record 2. The following is an example of a typical data file:

```
12 1 1
 1 X
 2 Y
```

```

3 Z
4 TY*BZ-TZ*BY
5 TZ*BX-TX*BZ
6 TX*BY-TY*BX
0 [METRE]
0.10000 0.26000 0.12625 0.49232E-03 -0.12105E-03 0.13192E-03
0.20000 0.26000 0.12625 0.49232E-03 0.12105E-03 0.13192E-03
0.30000 0.26000 0.12625 0.16246E-02 -0.10484E-03 0.16246E-02
0.40000 0.26000 0.12625 0.16246E-02 0.10484E-03 0.16246E-02
0.50000 0.26000 0.12625 0.13180E-03 -0.11982E-03 0.49189E-03
0.60000 0.26000 0.12625 0.13180E-03 0.11982E-03 0.49189E-03
0.70000 0.26000 0.12625 -0.13180E-03 -0.11982E-03 0.49189E-03
0.80000 0.26000 0.12625 -0.13180E-03 0.11982E-03 0.49189E-03
0.90000 0.26000 0.12625 -0.16246E-02 -0.10484E-03 0.16246E-02
1.00000 0.26000 0.12625 -0.16246E-02 0.10484E-03 0.16246E-02
1.10000 0.26000 0.12625 -0.49232E-03 -0.12105E-03 0.13192E-03
1.20000 0.26000 0.12625 -0.49232E-03 0.12105E-03 0.13192E-03

```

## TRACK files

The post processor **TRACK** command and the SCALA program store the trajectory coordinates in a binary unformatted file. There is no indication of the number of tracks in the file; the data must be read with end and error control to detect when all data has been read. The contents of the file is as follows:

Record 1: 20 `INTEGERs`. In track files created by the post processor, only the first value is important. It contains the number of points in the track, `NSTEP`.

Record 2: 20 `DOUBLE PRECISION` values, `RTRAK`.

`RTRAK(1)` is the current associated with the track.

`RTRAK(2)` is the mass.

`RTRAK(3)` is the charge.

`RTRAK(4)` is the step-length

`RTRAK(5-20)` can be ignored.

Record 3: `NSTEP` `DOUBLE PRECISION` values, x coordinates of the track points.

Record 4: `NSTEP` `DOUBLE PRECISION` values, y coordinates of the track points.

Record 5: `NSTEP` `DOUBLE PRECISION` values, z coordinates of the track points.

Record 6: `NSTEP` `DOUBLE PRECISION` values, x-component of the velocity.

Record 7: `NSTEP` `DOUBLE PRECISION` values, y-component of the velocity.

Record 8: `NSTEP` `DOUBLE PRECISION` values, z-component of the velocity.

On UNIX systems, if `NSTEP` is greater than 500, records 3 to 8 are blocked into sub-records which contain 500 `DOUBLE PRECISION` words.

## The **ACTIVATE** Command

---

### Summary

Make an OPERA-3d database available to the program.

### Icon



### Menu Routes:

File↓  
 Open  
 Recent files

### Command Line Parameters:

Command	<b>ACTIVATE</b>	
Parameter	Default	Function
<b>FILE</b>	<i>none</i>	Name of database file.
<b>XORIGIN</b>	0	X-coordinate of the coordinate system origin.
<b>YORIGIN</b>	0	Y-coordinate of the coordinate system origin.
<b>ZORIGIN</b>	0	Z-coordinate of the coordinate system origin.
<b>THETA</b>	0	Euler angle defining the coordinate system.
<b>PHI</b>	0	Euler angle defining the coordinate system.
<b>PSI</b>	0	Euler angle defining the coordinate system.
<b>SYMMETRY</b>	1	Rotational symmetry around local Z axis.
<b>RXY</b>		Reflection in local XY plane.
		<b>NO</b> No reflection.
		<b>YES</b> Reflection with same sign of scalar potential.
<b>RYZ</b>	<b>NO</b>	Reflection in local YZ plane.
		<b>YES</b> Reflection with same sign of scalar potential.
		<b>INVERSE</b> Reflection with inverse sign of scalar potential.

Command	<b>ACTIVATE</b> (continued)	
<b>RZX</b>	<b>NO</b>	Reflection in local ZX plane.
	<b>NO</b>	No reflection.
	<b>YES</b>	Reflection with same sign of scalar potential.
	<b>INVERSE</b>	Reflection with inverse sign of scalar potential.
<b>CASE</b>	1	Simulation number or * for the last simulation.

### Notes

The **ACTIVATE** command opens and checks a database file created by one of the analysis programs. The parameter **FILE** sets the file name. If no file name extension is given an extension of *op3* is assumed.

The **ACTIVATE** command also allows a local coordinate system and symmetry codes to be set. The local coordinate system is defined by its origin (**XORIGIN**, **YORIGIN**, **ZORIGIN**) and Euler angles (**THETA**, **PHI**, **PSI**). This repositions the whole model in space as if it had been defined in this position.

The symmetry codes can be used to create the complete model from the section which was analysed by rotation in the local Z axis and reflection in the local coordinate planes. Three types of symmetry are available.

- **SYMMETRY=n** creates *n* copies by rotating through  $360/n$  degrees around the local Z axis. The sign of the parameter **SYMMETRY** determines the direction of the field in the copies of the model. If **SYMMETRY** is negative the field direction is reversed in alternate copies.
- **Ruv=YES** should be used when the field normal to the local *uv* plane is zero. The program reflects the geometry in the *uv* plane of the local system and inverts the sign of the normal field in the reflected copy.
- **Ruv=INVERSE** should be used when the tangential field in the local *uv* plane is zero. The program reflects the geometry in the *uv* planes of the local system and inverts the sign of the tangential field in the reflected copy.

The field reflections apply to the principal field of the simulation, i.e. **H** for magnetic field problems (CARMEN, ELEKTRA, SOPRANO and TOSCA) and **E** for electric field problems (SCALA and TOSCA).

For databases with more than one simulation, the **CASE** parameter should be set to choose which simulation should be activated. **CASE=\*** indicates the last simulation in the database. After activating a database, the system variable **CASES** is updated with the number of simulations in the database.

Up to 100 files can be active at a time, each with its own coordinate system. In this way devices which form a complicated system, but which can be analysed separately can be linked together for post processing. The **SEARCH** parameter on the **SET** command controls the way in which multiple active database files are searched for field points (see [page 5-85](#)).

In menu mode, the **ACTIVATE** command is automatically followed by the **LOAD** command.

## The **ARC** Command

---

### Summary

Calculate fields along a circular arc defined by centre and end points.

### Icon



### Menu Route:

Fields↓  
Fields on an arc

### Command Line Parameters:

Command	<b>ARC</b>	
Parameter	Default	Function
<b>X1</b>	<i>none</i>	X-coordinate of the first point on the arc.
<b>Y1</b>	<i>none</i>	Y-coordinate of the first point on the arc.
<b>Z1</b>	<i>none</i>	Z-coordinate of the first point on the arc.
<b>X2</b>	<i>none</i>	X-coordinate of the last point on the arc.
<b>Y2</b>	<i>none</i>	Y-coordinate of the last point on the arc.
<b>Z2</b>	<i>none</i>	Z-coordinate of the last point on the arc.
<b>XC</b>	0	X-coordinate of the centre of curvature.
<b>YC</b>	0	Y-coordinate of the centre of curvature.
<b>ZC</b>	0	Z-coordinate of the centre of curvature.
<b>NP</b>	10	Number of steps between the first and last points, i.e. <b>NP</b> +1 points.

### Notes

The **ARC** command evaluates field quantities along a circular arc (Figure 5.1, on page 5-66) for use by the **FIT** and **PLOT** commands (see page 5-50 and page 5-73). Its function is similar to that of the **CIRCLE** command (see page 5-33). For each field point all the currently available system variables are calculated and stored (“System Variables” on page 5-10).

The arc is specified by its end points (**X1,Y1,Z1** and **X2,Y2,Z2**) and centre of curvature (**XC,YC,ZC**). The end points and centre of curvature must not be colin-

ear, otherwise the plane of the arc cannot be determined. The minor arc is always chosen. Major arcs can be specified with the **CIRCLE** command.

The positions of the end and centre points are affected by any local coordinate system defined with the **SET** command (page 5-85). The vector field quantities are evaluated with respect to the Global Coordinate System for the active file.

The field quantities are evaluated at **NP**+1 points along the arc.

It is possible to store the evaluated field quantities in a file using the **TABLE** command (page 5-97).

## The **ARITHMETIC** Command

---

**Summary**                      Combine data in two table files and creates a third.

**Menu Route:**                Tables↓  
   Arithmetic on table files

**Command Line Parameters:**

Command	<b>ARITHMETIC</b>	
Parameter	Default	Function
<b>FL1</b>	<i>none</i>	Name of first input data file.
<b>FL2</b>	<i>none</i>	Name of second input data file.
<b>FL3</b>	<i>none</i>	Name of output data file.
<b>VALUE</b>	<b>VAL1+VAL2</b>	Expression for values on output data file.

**Notes**                         The **ARITHMETIC** command combines the values in two *table* files and creates a new file of the same type. The format of the files is described in section “**TABLE Files**” on page 5-18.

The two input files, **FL1** and **FL2**, must contain the same number of records in the same configuration. For example, if they both represent points on a 2-dimensional patch, they must have the same numbers of points in the two directions given on the first record. They must also contain the same number of columns and the same units.

The output file, **FL3**, is created using the numbers of fields points, the column headings (Record type 2) and the units (Record type 3) from **FL1**. For each line of numerical data (Record type 4) the first three entries are assumed to be field point coordinates and are copied from **FL1**. For each subsequent column, the value from **FL1** is assigned to variable **VAL1** and the value from **FL2** to **VAL2** and the result of evaluating the expression given by **VALUE** is written to **FL3**.

## The **BHDATA** Command

---

**Summary** Plot graphs of BH characteristics used in analysis.

**Menu Route:** File↓  
Graph BH data

### **Command Line Parameters:**

Command	<b>BHDATA</b>	
Parameter	Default	Function
<b>MATERIAL</b>	1	Material name or number.
<b>DIRECTION</b>	<b>X</b>	Direction for anisotropic materials ( <b>X</b> , <b>Y</b> or <b>Z</b> ).

The **BHDATA** command displays the material characteristics used by the analysis programs. The parameter, **MATERIAL**, specifies the material name or number. A list of material names can be obtained with **MATERIAL=!**.

For isotropic or laminated materials, one curve is displayed; for anisotropic materials, the curve for the specified coordinate **DIRECTION** is displayed.

## The **BODY** Command

---

**Summary** Calculate Lorentz forces in conductors.

**Icon**



**Menu Route:** Integrals↓  
Lorentz forces in conductors

**Command Line Parameters:**

Command	<b>BODY</b>	
Parameter	Default	Function
<b>N1</b>	2	Number of gauss points in direction 1.
<b>N2</b>	2	Number of gauss points in direction 2.
<b>N3</b>	2	Number of gauss points in direction 3.
<b>X0</b>	0	X coordinate of point of action for torque.
<b>Y0</b>	0	Y coordinate of point of action for torque.
<b>Z0</b>	0	Z coordinate of point of action for torque.
<b>TAVERAGE</b>	<b>YES</b>	Time-average switch.
		<b>NO</b> Calculate integrals at time of <b>SET</b> command.
		<b>YES</b> Calculate time-average integrals.

**Notes**

The **BODY** command integrates the body forces and torques on the source conductors using  $\mathbf{J} \times \mathbf{B}$ . Each section of each conductor is represented by an 8 or 20 node finite element. Gaussian quadrature integration with a selectable number of Gauss points (in the range 1 to 10, or values 16 or 32) is used to calculate the forces.

The command operates as a 2-stage process:

1. form a list of conductors using the **CONDUCTOR** command (page 5-39). An empty list implies all conductors.

2. integrate forces using the **BODY** command.

The parameters **N1**, **N2** and **N3** specify the numbers of Gauss points in each direction. **N3** is the number of Gauss points in the direction of current flow.

The torque calculation uses the parameters **X0**, **Y0** and **Z0** to define the fixed point.

Unless **BODY** is used for conductor only problems it is essential that the conductors are completely enclosed in the finite element mesh. If the mesh does not completely contain the conductor the **SYMMETRY** and reflection parameters of the **ACTIVATE** command must be used to recreate the complete model (page 5-21).

The system variables **FX**, **FY** and **FZ** and **TORQX**, **TORQY** and **TORQZ** are updated with the values of force and torque.

The program calculates the force acting at the centroid of each section of each conductor. The coordinates and force on each section are listed in the file *Opera3d\_Post\_nn.lp*. The total force on each conductor and the total force on all conductors are displayed and listed in the file. If a conductor has reflections or symmetries, all copies are treated as one conductor.

Users should critically examine the forces to ensure that the net force on the conductor set has not arisen from field cancellation errors. The accuracy of force calculations can be enhanced by use of the integral conductor field recovery option (**SET COIL=INTEGRATE**, page 5-85). This option uses the values of **CURD** and **TOLERANCE** in the conductor definition which can be reset using the **CONDUCTOR** command (page 5-39). Changing **CURD** allows conductors to be switched on or off during the force calculations in order to find the self and mutual components of the force. Changing **TOLERANCE** further affects the accuracy of the conductor field calculations.

In steady-state alternating current problems, the force is a function of time with the form

$$F_x = A + B \cos 2\omega t + C \sin 2\omega t \quad (5.1)$$

The values **B** and **C** have little meaning on their own. The time-average value, **A** is the value commonly required. This can be calculated directly using the **+TAV-  
Erage** option.

The values of **B** and **C** can be found by setting the times to 0, 45 and 90 (see “**The SET Command**” on page 5-85), to give values of  $F_x$  at each time:  $F_0$ ,  $F_{45}$  and  $F_{90}$  with the **-TAVErage** option.

$$\begin{aligned}
 A &= \frac{F_0 + F_{90}}{2} \\
 B &= \frac{F_0 - F_{90}}{2} \\
 C &= F_{45} - A
 \end{aligned}
 \tag{5.2}$$

The following commands can be used to achieve this:

- Example - time-average  $F_x$ :

```

set time=0
body -tave
$ cons #fx0 fx
set time=45
body
$ cons #fx45 fx
set time=90
body
$ cons #fx90 fx
$ para #fxa 0.5*(#fx0+#fx90)
$ para #fxb 0.5*(#fx0-#fx90)
$ para #fxc #fx45-#fxa

```

The values  $F_0$ ,  $F_{45}$  and  $F_{90}$  have little meaning on their own.

## The **CARTESIAN** Command

---

### Summary

Calculate fields over a patch specified in xyz coordinates.

### Icon



### Menu Route:

Fields↓  
Fields on a cartesian a patch

### Command Line Parameters:

Command	<b>CARTESIAN</b>	
Parameter	Default	Function
<b>X1</b>	0	X-coordinate of the first corner of the surface.
<b>Y1</b>	0	Y-coordinate of the first corner of the surface.
<b>Z1</b>	0	Z-coordinate of the first corner of the surface.
<b>X2</b>	<i>none</i>	X-coordinate of the second corner of the surface.
<b>Y2</b>	<i>none</i>	Y-coordinate of the second corner of the surface.
<b>Z2</b>	<i>none</i>	Z-coordinate of the second corner of the surface.
<b>X3</b>	<i>none</i>	X-coordinate of the third corner of the surface.
<b>Y3</b>	<i>none</i>	Y-coordinate of the third corner of the surface.
<b>Z3</b>	<i>none</i>	Z-coordinate of the third corner of the surface.
<b>X4</b>	<i>none</i>	X-coordinate of the fourth corner of the surface.
<b>Y4</b>	<i>none</i>	Y-coordinate of the fourth corner of the surface.
<b>Z4</b>	<i>none</i>	Z-coordinate of the fourth corner of the surface.
<b>X5</b>	<i>none</i>	X-coordinate of the mid-point on side 1.
<b>Y5</b>	<i>none</i>	Y-coordinate of the mid-point on side 1.
<b>Z5</b>	<i>none</i>	Z-coordinate of the mid-point on side 1.
<b>X6</b>	<i>none</i>	X-coordinate of the mid-point on side 2.
<b>Y6</b>	<i>none</i>	Y-coordinate of the mid-point on side 2.
<b>Z6</b>	<i>none</i>	Z-coordinate of the mid-point on side 2.
<b>X7</b>	<i>none</i>	X-coordinate of the mid-point on side 3.
<b>Y7</b>	<i>none</i>	Y-coordinate of the mid-point on side 3.
<b>Z7</b>	<i>none</i>	Z-coordinate of the mid-point on side 3.

Command	<b>CARTESIAN</b> ( <i>continued</i> )	
Parameter	Default	Function
<b>X8</b>	<i>none</i>	X-coordinate of the mid-point on side 4.
<b>Y8</b>	<i>none</i>	Y-coordinate of the mid-point on side 4.
<b>Z8</b>	<i>none</i>	Z-coordinate of the mid-point on side 4.
<b>CORNERS</b>	4	Number of points defining surface: 4 or 8.
<b>N1</b>	10	Number of points on sides 1 and 3.
<b>N2</b>	10	Number of points on sides 2 and 4.

### Notes

The **CARTESIAN** command evaluates field quantities on 4 or 8-noded surface patches. The results can be displayed by the **MAP** command (page 5-68). For each field point all the currently available system variables are calculated and stored (see “System Variables” on page 5-10).

The patch is specified by its corner points (**X1,Y1,Z1**, **X2,Y2,Z2**, **X3,Y3,Z3** and **X4,Y4,Z4**) and optional mid-side points (**X5,Y5,Z5**, **X6,Y6,Z6**, **X7,Y7,Z7** and **X8,Y8,Z8**). The mid-side points are ignored if **CORNERS=4**, and included if **CORNERS=8**. Point 5 is on side 1 (point 1 to point 2); point 6 is on side 2 (point 2 to point 3); point 7 is on side 3 (point 3 to point 4); point 8 is on side 4 (point 4 to point 1).

The positions of the points are defined in the local coordinate system defined with the **SET** command (page 5-85). The field vectors are evaluated in the Global Coordinate System.

The field quantities are evaluated at **N1\*N2** points. The coordinates of points other than the corner and mid-side points are found by interpolation using 4 or 8-noded isoparametric two-dimensional finite element shape functions.

It is possible to store the evaluated field quantities in a file using the **TABLE** command (page 5-97).

## The CIRCLE Command

---

**Summary** Calculate fields along a circular arc specified by radius and angles.

**Icon**



**Menu Route:** Fields↓  
Fields on circle

**Command Line Parameters:**

Command	CIRCLE	
Parameter	Default	Function
RADIUS	none	Radius of circular arc.
TH1	0	Azimuthal-coordinate of the first point on the arc.
TH2	360	Azimuthal-coordinate of the last point on the arc.
ZC	0	Axial coordinate of arc.
NP	10	Number of steps between the first and last points, i.e. NP+1 points.

**Notes**

The CIRCLE command evaluates field quantities along a circular arc (Figure 5.1, on page 5-66) for use by the FIT and PLOT commands (page 5-50 and page 5-73). Its function is similar to that of the ARC command (page 5-24), except that it does allow arcs of 180 to 360 degrees to be defined. For each field point all the currently available system variables are calculated and stored (see page 5-10).

The circular arc is specified by its radius (RADIUS), the azimuthal coordinates of its end points (TH1 and TH2) and the axial coordinate (ZC). The plane of the arc is the local XY plane through local coordinates (0, 0, ZC), using the local coordinate system defined with the SET command (see page 5-85). The vector field quantities are evaluated with respect to the Global Coordinate System for the active file. The field quantities are evaluated at NP+1 points along the arc.

It is possible to store the evaluated field quantities in a file using the TABLE command (page 5-97).

## The **CLEAR** Command

---

**Summary** Clear all data and re-initialise all commands.

**Menu Route:** File↓  
Clear all data

**Command Line  
Parameters:**

Command	<b>CLEAR</b>
No Parameters	

**Notes**

The **CLEAR** command puts OPERA-3d back to the state it was in when it first started. It deletes all the data and user variables and sets all parameters back to their default values.

## The **COLOUR** Command

---

**Summary** Adjust the colours used for the display.

**Icon**



**Menu Route:** View↓  
Change colours

**Command Line Parameters:**

Command	COLOUR		
Parameter	Default	Function	
LABEL	TEXT	Label of colour to be redefined.	
		BACKGROUND	The background colour.
		CONDUCTORS	The colour of conductors.
		FIRSTCONTOUR	The colour of the minimum contour level.
		LASTCONTOUR	The colour of the maximum contour level.
		TEXT	The colour of text and axes.
		TRAJECTORIES	The colour of trajectories.
		VECTORINSIDE	The colour of the inside of vector cones.
		VECTOROUTSIDE	The colour of the outside of vector cones.
	<i>material_name</i>	The colour of a material.	
RED	<i>none</i>	Amount of red for colour.	
GREEN	<i>none</i>	Amount of green for colour.	
BLUE	<i>none</i>	Amount of blue for colour.	

**Notes**

The **COLOUR** command allows the user to adjust the colours used for parts of the display. The colours are identified by **LABEL** and the new values for the colour are given by numbers in the range 0 to 255.

- **FIRSTCONTOUR** and **LASTCONTOUR**: The program interpolates colours between the colours used for the extreme values.
- **CONDUCTORS** and materials: When a database is loaded the colours used by the Modeller or pre processor are also loaded (this applies to databases created by version 8.1 and above).

## The **COMBINE** Command

---

**Summary** Select both magnetic and electric fields for particle trajectory calculations.

**Menu Route:** Trajectories↓  
Use magnetic and electric fields

**Command Line Parameters:**

Command	<b>COMBINE</b>	
Parameter	Default	Function
<b>BOTH</b>	<b>NO</b>	Combine magnetic and electric fields for particle trajectories: <b>YES</b> or <b>NO</b> .
<b>SYMMETRY</b>	1	Rotational symmetry of additional field around local Z axis.
<b>RXY</b>	<b>NO</b>	<b>NO</b> No reflection.
		<b>YES</b> Field parallel to XY.
		<b>INVERSE</b> Field normal to XY.
<b>RYZ</b>	<b>NO</b>	<b>NO</b> No reflection.
		<b>YES</b> Field parallel to YZ.
		<b>INVERSE</b> Field normal to YZ.
<b>RZX</b>	<b>NO</b>	<b>NO</b> No reflection.
		<b>YES</b> Field parallel to ZX.
		<b>INVERSE</b> Field normal to ZX.
<b>MAGSCALE</b>	1	Scaling factor for magnetic field.
<b>ELESCALE</b>	1	Scaling factor for electric field.

**Notes** The **COMBINE** command switches combined field particle tracking on or off. Before it is used, a TOSCA or SCALA database containing electric and magnetic fields must be activated and loaded. Additional magnetic fields are used in the SCALA program for particle trajectory calculations. They can exist as a result of conductor data in the SCALA database file or can be added to the database using the **TABLE** command (page 5-97). The same **TABLE** command method can also

be used to add electric or magnetic fields to TOSCA databases which already contains the fields of the other type.

The geometric symmetry of the electrostatic and magnetostatic models should be the same. However, the field symmetry will often be different. If rotational and reflection symmetry is necessary on the **ACTIVATE** command for the model already activated, then the appropriate additional field symmetry should be specified on the **COMBINE** command using parameters **SYMMETRY**, **RXY**, **RYZ** and **RZX**.

The electric and magnetic fields can be scaled using the factors **MAGSCALE** and **ELESCALE**.

After switching the combined tracking option on, the **TRACK** command can be used in the usual way.

## The **CONDUCTOR** Command

---

### *Summary*

Define, erase, modify, list, import and export conductors.

### *Icons*



New solenoid



New racetrack



New bedstead



New helical end



New constant perimeter end



New straight bar



New arc



New 8-node brick



New 20-node brick



Toggle conductor picking



Modify selected conductors



Erase selected conductors



List conductor data

### *Menu Route:*

Conductors

**Command Line  
Parameters:**

Command	CONDUCTOR		
Parameter	Default	Function	
<b>ACTION</b>	<b>ADD</b>	Create list of conductors or integrate forces:	
		<b>ADD</b>	Add conductor(s) to list.
		<b>DEFAULT</b>	Set default values to match selected conductors.
		<b>DEFINE</b>	Define a new conductor.
		<b>DEFRESET</b>	Reset default values to match last conductor defined.
		<b>ERASE</b>	Erase selected conductors.
		<b>EXPORT</b>	Export conductors to a data file.
		<b>IMPORT</b>	Import conductors from a data file.
		<b>LIST</b>	List conductor data.
		<b>MODIFY</b>	Modify selected conductors.
		<b>PICK</b>	Add or remove conductor from list.
		<b>REMOVE</b>	Remove conductor(s) from list.
		<b>RESET</b>	Empty list of conductors.
		<b>STARTPICK</b>	Switch on picking facility.
<b>STOPPICK</b>	Switch off picking facility.		
<b>LABEL</b>	<b>ALL_CONDUCTORS</b>	Conductors to be added or removed from selection list:	
		<i>number</i>	Conductor number.
		<i>drive_label</i>	Drive label.
		<b>ALL_CONDUCTORS</b>	All conductors.

Command	CONDUCTOR (continued)	
Parameter	Default	Function
TYPE	SOLENOID	Conductor type: SOLENOID, RACE-TRACK, BEDSTEAD, HELICALEND, CPEND, STRAIGHT, ARC, BR8 or BR20.
XCENTRE		X coordinate of origin of local system 1.
YCENTRE		Y coordinate of origin of local system 1.
ZCENTRE		Z coordinate of origin of local system 1.
THETA1		Euler angle $\theta$ (local system 1).
PHI1		Euler angle $\phi$ (local system 1).
PSI1		Euler angle $\psi$ (local system 1).
X0		X coordinate of origin of local system 2.
Y0		Y coordinate of origin of local system 2.
Z0		Z coordinate of origin of local system 2.
T		Euler angle $\theta$ (local system 2).
P		Euler angle $\phi$ (local system 2).
S		Euler angle $\psi$ (local system 2).
IRXY		Reflection code in xy plane of local system 1.
IRYZ		Reflection code in yz plane of local system 1.
IRZX		Reflection code in zx plane of local system 1.
SYMMETRY		Symmetry code.
CURD		Current density.
TOLERANCE		Tolerance on flux density (negative for single filament approximation).
PHASE		Drive label.
X1		X coordinate of corner of conductor cross section.
Y1		Y coordinate of corner of conductor cross section.
X2		X coordinate of corner of solenoid cross section.
Y2		Y coordinate of corner of solenoid cross section.
X3		X coordinate of corner of solenoid cross section.

Command	<b>CONDUCTOR</b> ( <i>continued</i> )	
Parameter	Default	Function
<b>Y3</b>		Y coordinate of corner of solenoid cross section.
<b>X4</b>		X coordinate of corner of solenoid cross section.
<b>Y4</b>		Y coordinate of corner of solenoid cross section.
<b>CU1</b>		Curvature of cross section of solenoid (points 1 to 2).
<b>CU2</b>		Curvature of cross section of solenoid (points 2 to 3).
<b>CU3</b>		Curvature of cross section of solenoid (points 3 to 4).
<b>CU4</b>		Curvature of cross section of solenoid (points 4 to 1).
<b>A</b>		Thickness of conductor in x or radial direction.
<b>B</b>		Width of conductor in y or azimuthal direction.
<b>H1</b>		Length of straight section.
<b>H2</b>		Length of upright ( <b>BEDSTEAD</b> ) Local Z coordinate of midpoint of cross-over ( <b>HELIX</b> ).
<b>R1</b>		Radius: inner radius of arc ( <b>RACE-TRACK</b> , <b>BEDSTEAD</b> , <b>ARC</b> ). Radius of cylinder ( <b>HELIX</b> , <b>CPEND</b> ).
<b>R2</b>		Radius: inner radius of arc ( <b>BEDSTEAD</b> ). Width of cross-over ( <b>HELIX</b> ). Radius of generating cylinder ( <b>CPEND</b> ).
<b>PHI</b>		Angle of <b>ARC</b> .
<b>ALPHA</b>		Angle of straight from mid plane of cylinder ( <b>HELIX</b> and <b>CPEND</b> ).
<b>BETA</b>		Angle of end of helix ( <b>HELIX</b> ), or cutter ( <b>CPEND</b> ).
<b>FIT</b>		Fit of straight section to cylinder: <b>TANGENTIAL</b> or <b>FITTING</b> ( <b>CPEND</b> ).
<b>XB<sub>n</sub></b>		X coordinate of node of 8 or 20 node brick conductor ( $1 \leq n \leq 20$ ).

Command	<b>CONDUCTOR</b> (continued)	
Parameter	Default	Function
<b>YB<sub>n</sub></b>		Y coordinate of node of 8 or 20 node brick conductor ( $1 \leq n \leq 20$ ).
<b>ZB<sub>n</sub></b>		Z coordinate of node of 8 or 20 node brick conductor ( $1 \leq n \leq 20$ ).
<b>FILE</b>	none	Name of file for <b>IMPORT</b> or <b>EXPORT</b>

### Notes

This command controls the definition, modification, import and export of conductors. The operation of the command is controlled by the **ACTION** parameter:

- **ACTION=DEFINE**: defines a new conductor. The definition of the parameters and how they apply to each conductor shape is given in Chapter 4 (see “**The CONDUCTOR Command**” on page 4-25).
- **ACTION=ERASE**: erases a list of conductors. Before conductors can be erased the list must be formed using actions **RESET**, **ADD**, **PICK** and **REMOVE** (see below). After conductors have been erased the remaining conductors are renumbered to form a contiguous set starting at 1 and the selection list is emptied.
- **ACTION=MODIFY**: modifies a list of conductors. Before conductors can be modified the list must be formed using actions **RESET**, **ADD PICK**, and **REMOVE** (see below). The default values of the conductor parameters should also be set to correspond to the conductors selected using **ACTION=DEFAULT**. After modifying, the default values can be returned to correspond to the last conductor defined using **ACTION=DEFRESET**.

It is possible to set new values for parameters to expressions involving the old values, for example to double the current densities. The command sequence to do this to all conductors would be:

```
CONDUCTOR ACTION=ADD LABEL=ALL_CONDUCTORS
CONDUCTOR ACTION=DEFAULT
CONDUCTOR ACTION=MODIFY CURD=CURD*2
CONDUCTOR ACTION=DEFRESET
```

- **ACTION=DEFAULT**: sets the default values to correspond to the selected list of conductors in preparation for modification.
- **ACTION=DEFRESET**: resets the default values to correspond to the last conductor defined after using **ACTION=MODIFY**.
- **ACTION=LIST**: lists conductor data. If there is a selected list of conductors, only those conductors will be listed. Otherwise, all conductors will be listed.
- **ACTION=IMPORT**: reads conductor data from a conductor data file. Conductor data files are compatible with those created by the pre processor **CON-**

**DUCTOR** sub-command **WRITE** (page 4-49) and the Modeller command **EXPORT** (page 3-55).

- **ACTION=EXPORT**: writes conductor data to a file. If there is a selected list of conductors, only those conductors will be included. Otherwise, all conductors will be included.
- **ACTION=ADD**: adds conductors to the selected list by label. Labels can be conductor numbers, drive labels or **ALL\_CONDUCTORS**.
- **ACTION=REMOVE**: removes conductors from the selected list by label. Labels can be conductor numbers, drive labels or **ALL\_CONDUCTORS**.
- **ACTION=STARTPICK** switches on conductor picking.
- **ACTION=PICK** adds conductor number given by **LABEL** to the list if it is not already in the list, or removes it if already there.

This command is automatically generated by pointing at the appropriate conductor and double-clicking the left mouse button.

- **ACTION=STOPPICK** switches off conductor picking.
- **ACTION=RESET**: empties the selected list of conductors.

## The **END** Command

---

**Summary**            End the post processor

**Menu Route:**        File↓  
                          Exit

**Command Line  
Parameters:**

Command	<b>END</b>
No Parameters	

**Notes**                The **END** command stops the OPERA-3d post processor. All data files are closed.

## The ENERGY Command

---

### Summary

Calculate volume integrals to obtain stored energy, power loss and Lorentz forces.

### Icon



### Menu Route:

Integrals↓  
Energy, power & force

### Command Line Parameters:

Command	ENERGY		
Parameter	Default	Function	
ACTION	INTEGRATE	Create list of volumes or integrate:	
		ADD	Add volume(s) to list.
		INTEGRATE	Integrate.
		REMOVE	Remove volume(s) from list.
		RESET	Empty list of volumes.
LABEL	ALL_VOLUMES	volumes to be added or removed from list:	
		<i>material</i>	Material name.
		<i>label</i>	Volume label.
		ALL_VOLUMES	All volumes.
TAVERAGE	YES	Time-average switch.	
		NO	Calculate integrals at time of SET command.
		YES	Calculate time-average integrals.

Command	ENERGY (continued)			
Parameter	Default	Function		
ADAPTIVE	NO	Adaptive integration switch.		
		<table border="1"> <tr> <td>NO</td> <td>Use 8 gauss-points in each element.</td> </tr> <tr> <td>YES</td> <td>Use up to 216 gauss-points in each element.</td> </tr> </table>	NO	Use 8 gauss-points in each element.
NO	Use 8 gauss-points in each element.			
YES	Use up to 216 gauss-points in each element.			
MULTIPOLE	NO	Calculate multipole moments: YES or NO.		

### Notes

The **ENERGY** command integrates the stored energy, Lorentz force on induced currents, power loss and volume in the whole problem space or in a labelled set of elements and updates the following system variables:

Integrals	
Variable	Integrand
ENERGY	$\frac{1}{2} \mathbf{B} \cdot \mathbf{H}$ (magnetics)
	$\frac{1}{2} \mathbf{D} \cdot \mathbf{E}$ (electrostatics)
COENERGY	$\int (\mathbf{B} \cdot d\mathbf{H})$ (non-linear magnetostatics)
FX	$J_y B_z - J_z B_y$ (eddy currents)
FY	$J_z B_x - J_x B_z$ (eddy currents)
FZ	$J_x B_y - J_y B_x$ (eddy currents)
POWER	$ \mathbf{J} ^2 / \sigma$ (current flow and eddy currents)
VOLUME	1

In non-linear magnetostatics the stored energy can be found using the expression  $2 * \mathbf{ENERGY} - \mathbf{COENERGY}$ . In steady-state ac with complex permeability, the hysteresis contributions to energy and power are also calculated and displayed.

The command operates as a 2-stage process:

- form a list of volumes. Initially all volumes are in the list.
  - ACTION=RESET** empties the list.
  - ACTION=ADD** adds volumes to the list by **LABEL**.
  - ACTION=REMOVE** removes volumes from the list by **LABEL**.

- Labels can be material names, volume labels including element and potential types and user labels or **ALL\_VOLUMES**.
2. integrate forces using **ACTION=INTEGRATE**.

The volume integral of other expressions of system variables can be calculated using the **VOLUME** command (page 5-116).

## Integration Method

The basis for the integration is the finite element mesh. The integrals are performed in each element using first-order gaussian quadrature. However, in reduced potential volumes if the coil field is calculated by integration or anywhere if the total field is calculated by integration, the first-order quadrature is insufficient to match the field variation in an element. Switching on adaptive integration (**+ADAPTIVE**) enables the program to use up to 9<sup>th</sup>-order gaussian quadrature in each element to increase the accuracy of the integrals. See “The **SET Command**” on page 5-85 for information about field calculation methods.

The **ENERGY** command includes rotated and reflected copies of the mesh specified by the **ACTIVATE** command (page 5-21) in its integration.

## Steady-state AC problems

In steady-state alternating current problems, the energy and power are functions of time with the form

$$E = A + B \cos 2\omega t + C \sin 2\omega t \quad (5.3)$$

The values  $B$  and  $C$  have little meaning on their own. The time-average value,  $A$  is the value commonly required. This can be calculated directly using **+TAVERAGE**.

The values of  $B$  and  $C$  can be found by setting the times to 0, 45 and 90 (page 5-85), to give values of  $E$  at each time:  $E_0$ ,  $E_{45}$  and  $E_{90}$  with the **-TAVERAGE** option.

$$\begin{aligned} A &= \frac{E_0 + E_{90}}{2} \\ B &= \frac{E_0 - E_{90}}{2} \\ C &= E_{45} - A \end{aligned} \quad (5.4)$$

The following commands can be used to achieve this:

- Example - time-average energy:

```
set time=0
ener -tave
$ cons #en0 energy
set time=45
ener
$ cons #en45 energy
set time=90
ener
$ cons #en90 energy
$ para #ena 0.5*(#en0+#en90)
$ para #enb 0.5*(#en0-#en90)
$ para #enc #en45-#ena
```

The values  $E_0$ ,  $E_{45}$  and  $E_{90}$  have little meaning on their own.

## Multipole Moments

**MULTIPOLE** moments can be calculated for all non-AIR materials in TOSCA magnetostatic simulations. The values are stored in a separate output file, with the name taken from the OPERA-3d database, but with extension *.mpole*. The volume, dipole, quadrupole and octupole moments are stored for all materials and for each material. Each moment value is given in two unit sets: amp metre<sup>n</sup> and gamma ft<sup>n</sup> (1 gamma = 10<sup>-9</sup> Tesla).

## The **FIT** Command

---

### Summary

Fit Fourier series to field values previously calculated along a line or calculate field values on a sphere and fit Legendre polynomials.

### Menu Routes:

Fields↓

Fit Fourier series to values

Fit Legendre polynomials to values

### Command Line

#### Parameters:

Command	<b>FIT</b>	
Parameter	Default	Function
<b>FILE</b>	<b>TEMP</b>	Name of the table file containing the field points and values. <b>FILE=TEMP</b> means do not use a file.
<b>TYPE</b>	<b>FOURIER</b>	Type of fitting.
		<b>FOURIER</b> <b>FOURIER</b> series fitting to <b>ARC</b> , <b>CIRCLE</b> or <b>LINE</b> values. <b>LEGENDRE</b> <b>LEGENDRE</b> polynomial fitting to values on a sphere around local coordinate system origin.
<b>COMPONENT</b>	<b>X</b>	Field component for fitting.
<b>PRINT</b>	<b>YES</b>	Print options.
		<b>LOG</b> Output to log file <i>opera.lp</i> . <b>SCREEN</b> Output to screen. <b>YES</b> Output to both <b>SCREEN</b> and <b>LOG</b> file. <b>NO</b> No output.
<b>RADIUS</b>	1	Radius of sphere for <b>LEGENDRE</b> polynomial fitting.
<b>ORDER</b>	10	Maximum degree of <b>LEGENDRE</b> polynomials.

**Notes**

The **FIT** command is used to fit **FOURIER** series or **LEGENDRE** polynomials to field values.

**FOURIER** series fitting uses field values previously calculated by the **ARC**, **CIRCLE** or **LINE** commands (page 5-24, page 5-33 and page 5-65). The values are assumed to span a complete cycle. The values can be those stored in the program (**FILE=TEMP**) or can be read in from a file by the **FIT** command. The structure of the files is given in section “**TABLE Files**” on page 5-18. If no file name extension is given the extension *table* is assumed.

**LEGENDRE** polynomial fitting first calculates values on the surface of a sphere which must be completely inside the problem space. It may be necessary to **ACTIVATE** the results with symmetry parameters in order to achieve this (page 5-21). The radius of the sphere is given by the **RADIUS** parameter, and its origin and orientation are defined by the local coordinate system of the **SET** command (page 5-85). The maximum order of polynomial can be set with the **ORDER** parameter; it is limited to 30. However for  $9 \leq \text{ORDER} \leq 14$ , coefficients up to order 14 will be given and for  $\text{ORDER} \geq 14$  coefficients up to order 30 will be given. The values at the field points used by the fitting algorithm can be displayed by the **MAP** commands (page 5-68).

If a file is not used (**FILE=TEMP**), expressions for the **COMPONENT** can use as variables any of the system variables which are currently available (see “**System Variables**” on page 5-10). If a table file is used, the component name and values are read from the 4<sup>th</sup> column of the file.

## The **GRAPH** Command

---

### Summary

Plot graphs from data in text files.

### Menu Route

File↓  
Graph data in text file

### Command Line Parameters

Command	<b>GRAPH</b>		
Parameter	Default	Function	
<b>FILE</b>	<i>none</i>	File name	
<b>XVALUE</b>	<b>COL1</b>	Expression for x-axis values.	
<b>XLABEL</b>	<b>X</b>	Label for x-axis.	
<b>XMINIMUM</b>	*	X-axis minimum value. Use * for automatic scaling.	
<b>XMAXIMUM</b>	*	X-axis maximum value. Use * for automatic scaling.	
<b>YVALUE</b>	<b>COL2</b>	Expression for y-axis values.	
<b>YLABEL</b>	<b>Y</b>	Label for y-axis.	
<b>YMINIMUM</b>	*	Y-axis minimum value. Use * for automatic scaling.	
<b>YMAXIMUM</b>	*	Y-axis maximum value. Use * for automatic scaling.	
<b>TITLE</b>	<i>none</i>	Additional title for line.	
<b>STYLE</b>	<b>AUTOMATIC</b>	Line style:	
		<b>AUTOMATIC</b>	Program chooses a different style for each graph drawn on the same axes.
		0	Solid line.
		>0	Broken line.

Command	GRAPH (continued)		
Parameter	Default	Function	
COLOUR	AUTOMATIC	Line colour:	
		AUTOMATIC	Program chooses a different colour for each graph drawn on the same axes.
		>0	Colour number.
SYMBOL	AUTOMATIC	Symbols at data points:	
		AUTOMATIC	Program chooses a different symbol for each graph drawn on the same axes.
		0	No symbol.
		>0	Symbols number.
ERASE	YES	Old graph erasure switch:	
		NO	New line drawn on existing axes.
		YES	Graphics window cleared and new axes drawn.
OPTION	VALUES	Display option:	
		DERIVATIVES	Display the first derivatives of the data
		INTERPOLATIONS	Display cubic-spline interpolations between data values.
		VALUES	Display straight lines between data values.

### Notes

The **GRAPH** command is a general purpose command for displaying graphs of data read from external files.

The parameters of the **GRAPH** command control the axes limits (**XMINIMUM**, **XMAXIMUM**, **YMINIMUM** and **YMAXIMUM**), the line **STYLE** and **COLOUR** and whether **SYMBOLS** should be displayed at the data points. For graphs with more than one line, the second and subsequent lines should be drawn with **ERASE**. A line **TITLE** can be specified for each line. The line titles appear in a legend at the bottom left corner of the graph.

The data format expected by the graphs command is flexible. The file should consist of up to 20 columns of numbers, with any number of values in each column. However, text can be embedded within the numerical data. Each line of the file is parsed into a maximum of 20 fields separated by spaces or commas. The fields are identified as character data or numerical data. If there is numerical data on a line it is kept; character data is ignored. The number of columns is given by the number of numerical data items on the first line which contains any numbers. If subsequent lines contain less numerical values, the number of columns is reduced.

The values for the x and y coordinates of the points plotted on the graphs can be calculated using expressions in terms of corresponding entries from the columns. The simplest use would be to use the first column for the x-coordinates and the second for the y-coordinates (**XVAL=COL1**, **YVAL=COL2**), but much more complicated expressions can be used. For example, the percentage difference between two columns could be calculated and displayed as a graph using **YVAL=100\*(COL3-COL2)/COL2**. The **ROW** number can also be used in expressions.

The data evaluated from the expressions **XVALUE** and **YVALUE** can be displayed in 3 ways:

- **OPTION=VALUE** shows straight lines between the evaluated data points.
- **OPTION=INTERPOLATIONS** uses cubic-spline interpolations between the data points.
- **OPTION=DERIVATIVES** shows the first derivative of the cubic-spline interpolations.

**INTERPOLATIONS** and **DERIVATIVES** can only be used if the expression for **XVALUE** results in a mono-tonically increasing set of values.

## The **GRID** Command

---

### Summary

Calculate field values on a grid of points and write a table file.

### Menu Route:

Tables↓

Table of fields values on a grid

### Command Line Parameters:

Command	GRID	
Parameter	Default	Function
<b>X0</b>	0	X-coordinate at corner of grid.
<b>Y0</b>	0	Y-coordinate at corner of grid.
<b>Z0</b>	0	Z-coordinate at corner of grid.
<b>DXG</b>	1	X increment between grid points
<b>DYG</b>	1	Y increment between grid points
<b>DZG</b>	1	Z increment between grid points
<b>NXG</b>	1	Number of points in X direction
<b>NYG</b>	1	Number of points in Y direction
<b>NZG</b>	1	Number of points in Z direction
<b>FILE</b>	<i>none</i>	Name of file to store values.
<b>BINARY</b>	<b>NO</b>	Binary file switch: <b>YES</b> or <b>NO</b> . <b>NO</b> implies a text file.
<b>F1</b>	<b>X</b>	Expression for values in column 1
<b>F2</b>	<b>Y</b>	Expression for values in column 2
<b>F3</b>	<b>Z</b>	Expression for values in column 3
<b>F4</b>	<i>none</i>	Expression for values in column 4
<b>F5</b>	<i>none</i>	Expression for values in column 5
<b>F6</b>	<i>none</i>	Expression for values in column 6
<b>F7</b>	<i>none</i>	Expression for values in column 7
<b>F8</b>	<i>none</i>	Expression for values in column 8
<b>F9</b>	<i>none</i>	Expression for values in column 9
<b>F10</b>	<i>none</i>	Expression for values in column 10
<b>F11</b>	<i>none</i>	Expression for values in column 11
<b>F12</b>	<i>none</i>	Expression for values in column 12

**Notes**

The **GRID** command is provided to facilitate an interface to other post processing programs. It calculates field values over a 1, 2 or 3 dimensional space defined by one corner (**X0, Y0, Z0**), the increments in each direction (**DXG, DYG, DZG**) and the number of points in each direction (**NXG, NYG, NZG**). The coordinates of the points are defined with respect to the local coordinate system of the **SET** command (page 5-85). The field points and components are output to a file in the Global Coordinate System.

Expressions for values in up to 12 columns can use any of the system variables which are currently available (see “**System Variables**” on page 5-10). In text files (**-BINARY**) table file format is used (see “**TABLE Files**” on page 5-18). Binary files (**+BINARY**) are written using grid file format (see “**GRID files**” on page 5-18.)

If no file name extension is given, the extension *table* is supplied for text files or *grid* for binary.

## The IDEAS Command

---

**Summary** Create or append to a SDRC I-DEAS Universal File.

**Menu Route:** Tables↓  
SDRC I-DEAS Unv file

**Command Line Parameters:**

Command	IDEAS		
Parameter	Default	Function	
<b>FILE</b>	<i>none</i>	Name of Universal File.	
<b>MODE</b>	<b>APPEND</b>	Mode of operation:	
		<b>APPEND</b>	Append results to an existing file.
		<b>CREATE</b>	Create a new file with nodes, elements, material names and results.
<b>TYPE</b>	<b>REAL</b>	Result type:	
		<b>COMPLEX</b>	Results from steady-state ac analysis
		<b>REAL</b>	Results from statics or transient analysis.
<b>BASIS</b>	<b>NODE</b>	Basis of results:	
		<b>ELEMENT</b>	Values at every node of every element
		<b>NODE</b>	Values at every node
<b>FIELD</b>	<b>SCALAR</b>	Field type:	
		<b>SCALAR</b>	Scalar field
		<b>VECTOR</b>	Vector field
<b>COMPONENT</b>	<b>X</b>	Expression for real part of scalar field values.	
<b>ICOMPONENT</b>	<b>Y</b>	Expression for imaginary part of scalar field values ( <b>TYPE=COMPLEX</b> )	
<b>VX</b>	<b>X</b>	Expression of real part of x-component of vector field.	

Command	<b>IDEAS</b> (continued)	
Parameter	Default	Function
<b>IVX</b>	<b>X</b>	Expression for imaginary part of x-component of vector field
<b>VY</b>	<b>Y</b>	Expression of real part of y-component of vector field.
<b>IVY</b>	<b>Y</b>	Expression for imaginary part of y-component of vector field
<b>VZ</b>	<b>Z</b>	Expression of real part of z-component of vector field.
<b>IVZ</b>	<b>Z</b>	Expression for imaginary part of z-component of vector field
<b>LABEL</b>	1	Analysis dataset label
<b>NAME</b>	<b>VF</b>	Analysis dataset name

### Notes

The **IDEAS** command creates or appends to an I-DEAS Universal file. When the file is created (**MODE=CREATE**) the finite element data is written first, followed by a results dataset. In **APPEND** mode, only the results dataset is written.

## Finite Element Data

The finite element data written to a Universal File consists of the following datasets:

Dataset Number	Dataset Name
151	Header
164	Units
1700	Material Database Header
1703	Material Database Property
1705	Material Database Variable
1710	Material Database Material
789	Physical Properties
2411	Nodes - Double Precision
2412	Elements

Dataset 164 specifies the length unit for the data. The 'Units description' includes the units of flux density, current density and electric field strength.

Dataset 1710 is repeated for each 'material', where a material is formed for each combination of material name, potential type and element type which exists in the

model, for example a typical Universal File Material Name would be IRON\_TOTAL\_QUADRATIC. The 'Material Number' and 'Material Name' are the only significant data in the dataset 1710.

## Results Data

Results data are written using dataset 2414 'Analysis Data'. The simplest form is a real scalar field at nodes (**TYPE=REAL**, **BASIS=NODE**, **FIELD=SCALAR**) which can be specified using one **COMPONENT** expression. At the other extreme is a complex vector field on elements, allowing discontinuity between each element and its neighbours (**TYPE=COMPLEX**, **BASIS=ELEMENT**, **FIELD=VECTOR**). This would be specified using 6 expressions, for the real and imaginary parts of the X, Y and Z components.

All other combinations of **TYPE**, **BASIS** and **FIELD** are also allowed, for example, to store the electric field strength in an electrostatic example, not allowing for any discontinuities:

```
type=real,basi=node,fiel=vect,vx=ex,vy=ey,vz=ez
```

## Limitations for Quadratic Elements

If the finite element model in the OPERA-3d database was created from a Universal File using the **IDEAS** command in the pre processor (page 4-119), and it also contains quadratic elements, the results should not be **APPENDED** to the original Universal File, since the positions and ordering of the quadratic nodes will not match. A new Universal File should be **CREATED** to contain the finite element model and results.

## Integration Commands

---

In many situations, the result which is required is calculated by integrating field quantities along a line, over a surface or in a volume. OPERA-3d post processor has several commands which can perform integration, some for specific purposes and others which allow more general facilities.

### Line Integrals

Line integrals are calculated by the **PLOT** command. The line should be defined and field quantities evaluated first by **ARC**, **CIRCLE** or **LINE**.

The **PLOT** command (page 5-73) draws a graph of any field component expression and also calculates the line integral.

### Surface Integrals

There are two types of surface integral:

- *Over a surface of the model:* The surface is the currently **SELECT**ed surface (page 5-80). It can be the surface of a material or potential type or a boundary condition surface.
  - The **INTEGRATE** command (page 5-62) integrates the Maxwell Stress tensor over the surface to calculate force and torque.
  - The **SURFACE** command (page 5-90) integrates any field component expression over the surface.
- *Over a general surface patch:* The surface should be defined and the field quantities evaluated first using the **CARTESIAN** or **POLAR** commands.
  - The **MAP** commands (page 5-68) display distribution of the field component expression and also calculate the area integral.

### Volume Integrals

Volume integrals can be performed over volumes selected by labels (including material names, potential names, element types, **NOTAIR** and **ALL\_VOLUMES**). The basis for the integration is the finite element mesh; higher order integration formulae are available for functions which vary rapidly over the elements.

- The **ENERGY** command (page 5-46) has built-in integrands to calculate energy, power-loss and Lorentz forces in the complete model including replications specified on the **ACTIVATE** command.
- The **VOLUME** command (page 5-116) can integrate any field component expression. Replications of the model are omitted from the integration.

The **BODY** command (page 5-28) integrates Lorentz forces over the volume of the conductors.

## The **INTEGRATE** Command

---

### Summary

Integrate Maxwell stress over selected surfaces to obtain forces on enclosed volume.

### Icon



### Menu Route:

Integrals↓  
Maxwell stress on selected surfaces

### Command Line Parameters:

Command	<b>INTEGRATE</b>	
Parameter	Default	Function
<b>X0</b>	0	X coordinate of point of action for torque.
<b>Y0</b>	0	Y coordinate of point of action for torque.
<b>Z0</b>	0	Z coordinate of point of action for torque.
<b>TAVERAGE</b>	<b>YES</b>	Time-average switch.
	<b>NO</b>	Calculate integrals at time of <b>SET</b> command.
	<b>YES</b>	Calculate time-average integrals.

### Notes

The **INTEGRATE** command integrates the Maxwell Stress tensor over the surface selected by the **SELECT** command (page 5-80) to calculate the force and torque. The parameters (**X0**, **Y0**, **Z0**) define the global coordinates of the point of action of the torque. The results are also stored in the system variables **FX**, **FY** and **FZ** and **TORQX**, **TORQY** and **TORQZ**.

For magnetic field simulations functions integrated are:

$$\begin{aligned}
 F_x &= \int_s \left[ \frac{1}{\mu} B_x (\mathbf{B} \cdot \mathbf{n}) - \frac{1}{2\mu} |\mathbf{B}|^2 n_x \right] ds \\
 F_y &= \int_s \left[ \frac{1}{\mu} B_y (\mathbf{B} \cdot \mathbf{n}) - \frac{1}{2\mu} |\mathbf{B}|^2 n_y \right] ds \\
 F_z &= \int_s \left[ \frac{1}{\mu} B_z (\mathbf{B} \cdot \mathbf{n}) - \frac{1}{2\mu} |\mathbf{B}|^2 n_z \right] ds
 \end{aligned} \tag{5.5}$$

For electrostatic field simulations functions integrated are:

$$\begin{aligned}
 F_x &= \int_s \left[ \frac{1}{\mu} D_x (\mathbf{D} \cdot \mathbf{n}) - \frac{1}{2\mu} |\mathbf{D}|^2 n_x \right] ds \\
 F_y &= \int_s \left[ \frac{1}{\mu} D_y (\mathbf{D} \cdot \mathbf{n}) - \frac{1}{2\mu} |\mathbf{D}|^2 n_y \right] ds \\
 F_z &= \int_s \left[ \frac{1}{\mu} D_z (\mathbf{D} \cdot \mathbf{n}) - \frac{1}{2\mu} |\mathbf{D}|^2 n_z \right] ds
 \end{aligned} \tag{5.6}$$

Better results can often be obtained if a layer of air is added to the selected surfaces before **INTEGRATE** to move the integration surface away from the material corners where the field may be singular (See “**THE SELECT Command**” on page 5-80.). In any case, on a material surface, the **INTEGRATE** command uses the values from the air side of the interface.

If the force required is acting on the coils, it is usually more accurate to select the surface of the reduced scalar potential volume containing the coils and **INTEGRATE** over that or use the **BODY** command (page 5-28),

The **MAP** command can also be used to integrate over more general surfaces. The functions necessary for calculating forces are given in terms of **\$ PARAMETER** commands in section “**Examples**” on page 5-16. The **SURFACE** command can be used to integrate other field quantities over the selected surface (page 5-90).

In steady-state alternating current problems, the force is a function of time with the form

$$F_x = A + B \cos 2\omega t + C \sin 2\omega t \tag{5.7}$$

The values  $B$  and  $C$  have little meaning on their own. The time-average value,  $A$  is the value commonly required. This can be calculated directly using **+TAVERAGE**.

The values of  $B$  and  $C$  can be found by setting the times to 0, 45 and 90 (page 5-85), to give values of  $F_x$  at each time:  $F_0$ ,  $F_{45}$  and  $F_{90}$  with the **-TAVERAGE** option.

$$\begin{aligned} A &= \frac{F_0 + F_{90}}{2} \\ B &= \frac{F_0 - F_{90}}{2} \\ C &= F_{45} - A \end{aligned} \tag{5.8}$$

The following commands can be used to achieve this:

- Example - time-average  $F_x$ :

```
set time=0
inte -tave
$ cons #fx0 fx
set time=45
inte
$ cons #fx45 fx
set time=90
inte
$ cons #fx90 fx
$ para #fxa 0.5*(#fx0+#fx90)
$ para #fxb 0.5*(#fx0-#fx90)
$ para #fxc #fx45-#fxa
```

The values  $F_0$ ,  $F_{45}$  and  $F_{90}$  have little meaning on their own.

## The **LINE** Command

---

**Summary** Calculate fields along a straight line.

**Icon**



**Menu Route:** Fields↓  
Fields on a straight line

**Command Line Parameters:**

Command	<b>LINE</b>	
Parameter	Default	Function
<b>X1</b>	0	X-coordinate of the first point on the line.
<b>Y1</b>	0	Y-coordinate of the first point on the line.
<b>Z1</b>	0	Z-coordinate of the first point on the line.
<b>X2</b>	<i>none</i>	X-coordinate of the last point on the line.
<b>Y2</b>	<i>none</i>	Y-coordinate of the last point on the line.
<b>Z2</b>	<i>none</i>	Z-coordinate of the last point on the line.
<b>NP</b>	10	Number of steps between the first and last points, i.e. <b>NP</b> +1 points.

**Notes**

The **LINE** command evaluates field quantities along a straight line (Figure 5.1) for use by the **FIT** and **PLOT** commands (page 5-50 and page 5-73). For each field point all the currently available system variables are calculated and stored (see “System Variables” on page 5-10).

The line is specified by its end points (**X1,Y1,Z1** and **X2,Y2,Z2**), the positions of which are affected by any local coordinate system defined with the **SET** command (page 5-85). The vector field quantities are evaluated with respect to the Global Coordinate System for the active file. The field quantities are evaluated at **NP**+1 points along the line.

It is possible to store the evaluated field quantities in a file using the **TABLE** command (page 5-97).

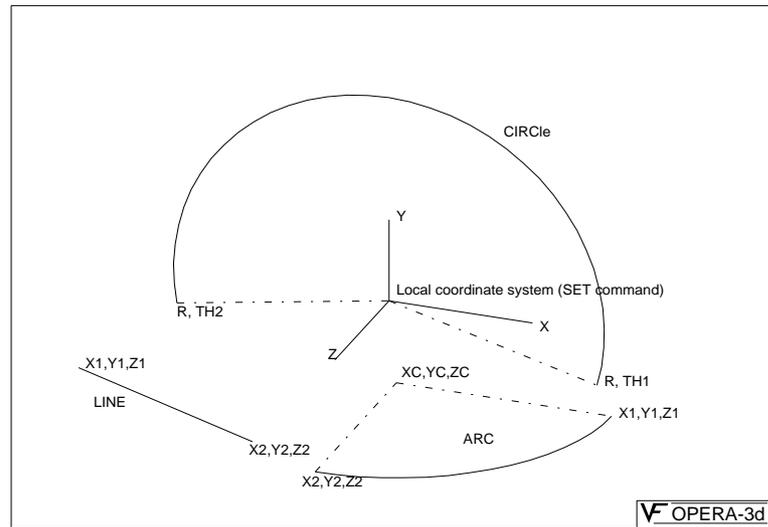


Figure 5.1 The **ARC**, **CIRCLE** and **LINE**

## The **LOAD** Command

---

**Summary** Nominate an active database for display of geometry and field calculations.

**Menu Route:** File↓  
 Open  
 Re-load active database  
 Close loaded database

**Command Line Parameters:**

Command	<b>LOAD</b>	
Parameter	Default	Function
<b>FILE</b>	<i>none</i>	Name or number of an active database file.

**Notes** The **LOAD** command makes an active database file resident. Files are activated by the **ACTIVATE** command (page 5-21). The **LOAD** command makes the data in an active file available for the display and field evaluation commands. The details of the loaded or resident file are shown in the Information Box at the right hand side of the display.

The parameter **FILE** sets the file name. If no file name extension is given an extension of *op3* is assumed. **FILE** can also specify the number of the active file. This can be useful if the same file has been activated more than once with different parameters on the **ACTIVATE** command. The active files can be listed with the **SHOW** command (page 5-88). If no file name or number is given the most recently activated file is assumed.

**LOAD FILE=0** can be used to close the currently loaded file so that it can be accessed by other programs.

The menu system automatically executes the **LOAD** command when a file is opened.

## The **MAP** Command

---

### Summary

Display contours of field vectors on 2-dimensional surface patches.

### Icon



### Menu Route:

Fields↓  
Contour or vector map

### Command Line Parameters:

Command	MAP	
Parameter	Default	Function
<b>FILE</b>	<b>TEMP</b>	Name of table file containing field points and values. <b>FILE=TEMP</b> means do not use a file.
<b>CONTOUR</b>	<b>ZONES</b>	Display <b>COMPONENT</b> as a contour map.
		<b>HISTOGRAM</b> Contours displayed as a histogram.
		<b>LINES</b> Contour lines.
		<b>NO</b> No contour map.
	<b>ZONES</b>	Coloured zone contours
<b>COMPONENT</b>	<b>X</b>	Field component for contours.
<b>MIN</b>	*	Minimum contour value. * for automatic setting.
<b>MAX</b>	*	Maximum contour value. * for automatic setting.
<b>VECTORS</b>	<b>NO</b>	Vector display switch.
		<b>YES</b> <b>VECTORS</b> displayed as cones with components specified by <b>VX</b> , <b>VY</b> and <b>VZ</b> .
<b>VX</b>	<b>X</b>	Expression for x-component of vectors.
<b>VY</b>	<b>Y</b>	Expression for y-component of vectors.
<b>VZ</b>	<b>Z</b>	Expression for z-component of vectors.

Command	MAP (continued)		
Parameter	Default	Function	
PRINT	NO	Print options.	
		YES	Output to log file.
		NO	No output.
HEIGHT	SIZE/3	Height of histogram.	
ERASE	YES	Multiple map options:	
		YES	Replace any existing map.
		NO	Add to existing maps.
LINES	10	Number of line contours.	

### Notes

The **MAP** command is used to display field values calculated with the **CARTESIAN**, **FIT TYPE=LEGENDRE** and **POLAR** commands (page 5-31, page 5-50 and page 5-77) as a contour map or as vectors. These can be values stored in the program (**FILE=TEMP**) or read in from a table file by the **MAP** command. The structure of the files is given in section “**TABLE Files**” on page 5-18. If no file name extension is given the extension *table* is assumed.

The map is overlaid on the three dimensional pictures of the geometry created by the **THREED** command (page 5-101). Any number of contour maps can be added to the 3d Viewer (**ERASE=NO**) or a map can replace any previously displayed (**ERASE=YES**). The maps and the scale of values can be temporarily removed from the display using the **WINDOW** command (page 5-119).

Two types of map are available:

- **CONTOURS** of a scalar **COMPONENT**. The contour map can appear at the location of the field points (**contour=zones** or **contour=lines**) or displaced in proportion to the component value as a 3d histogram (**contour=histogram**).  
**LINES** sets the number of contour lines.  
**HEIGHT** sets the height of the histogram. The default value gives a height of one third of the **THREED** command **SIZE** parameter.
- **VECTORS** of a field vector with components specified by **VX**, **VY** and **VZ**. **CONTOURS** and **VECTORS** can be displayed with one command.

If a file is not used (**FILE=TEMP**), expressions for the scalar **COMPONENT** and the components of the vectors, **VX**, **VY**, **VZ** can use as variables any of the system variables which are currently available (section “**System Variables**” on page 5-10). If a table file is used, the component name and values are read from the 4<sup>th</sup> column of the file; the vector components for columns 5, 6 and 7.

The values of the maximum and minimum contour values can be set with **MAX** and **MIN**. If either **MAX** or **MIN** is set to \* the value is calculated from the range of values to be displayed.

The lengths of the vectors can be scaled by using a scaling factor in the expressions for **VX**, **VY** and **VZ**. Automatic scaling can be achieved by using the system variable **MAXIMUM** from a previous **MAP** command where the **COMPONENT** contoured was the magnitude of the vectors to be scaled.

The **MAP** command also calculates the integral of the **COMPONENT**, and stores its value in the system variable **INTEGRAL**. The maximum and minimum values of **COMPONENT** are stored in system variables **MAXIMUM** and **MINIMUM**.

## The **MOUSE** Command

---

**Summary** Swap the functionality of middle and right mouse buttons.

**Icon**



**Menu Route:**

Options↓  
Toggle right mouse button

**Command Line Parameters:**

Command	<b>MOUSE</b>
No Parameters	

**Notes**

The **MOUSE** command swaps the functionality of the middle and right mouse buttons so that the software can be used with a 2-button mouse.

The default mouse functions in the 3d Graphics Window are:

- left: rotate
- middle: zoom
- right: translate

## The **PICTURE** Command

---

**Summary** Save the current display in a file or on the clipboard.

### Icons



### Menu Route:

File↓  
 Copy to file  
 Copy to clipboard

### Command Line Parameters:

Command	<b>PICTURE</b>		
Parameter	Default	Function	
<b>SAVE</b>	<b>NO</b>	<b>NO</b>	Copy file to clipboard
		<b>YES</b>	Save picture in a file
<b>FILENAME</b>		The name of the file to be saved	
<b>TYPE</b>	<b>PNG</b>	<b>PNG</b>	Type of image file format used to save the picture
		<b>BMP</b>	
		<b>XPM</b>	

### Notes

This command allows the current display to be stored on the clipboard or in a file.

If **SAVE=NO** the image is placed on the clipboard and can then be pasted into another application. Under UNIX, the image is placed on the X-Selection.

With **SAVE=YES**, the image is stored in the format specified in the given file.

An image can be printed directly using the **PRINT** command.

## The **PLOT** Command

---

### Summary

Plot a graph and integrate field values from **LINE**, **ARC** or **CIRCLE** commands.

### Icon



### Menu Route:

Fields↓  
Plot graph of field values

### Command Line Parameters:

Command	PLOT			
Parameter	Default	Function		
<b>FILE</b>	<b>TEMP</b>	Name of table file containing field points and values. <b>FILE=TEMP</b> means do not use a file.		
<b>COMPONENT</b>	<b>X</b>	Field component for graph.		
<b>YMIN</b>	*	Minimum value for Y-axis of graph. * for automatic setting.		
<b>YMAX</b>	*	Maximum value for Y-axis of graph. * for automatic setting.		
<b>ERASE</b>	<b>YES</b>	Erase screen switch:		
		<table border="0"> <tr> <td><b>NO</b></td> <td>Draw new graph on existing axes.</td> </tr> <tr> <td><b>YES</b></td> <td>Erase old picture first and draw new axes.</td> </tr> </table>	<b>NO</b>	Draw new graph on existing axes.
<b>NO</b>	Draw new graph on existing axes.			
<b>YES</b>	Erase old picture first and draw new axes.			
<b>PRINT</b>	<b>LOG</b>	Print options:		
		<b>LOG</b>	Output values to log file, <i>opera.lp</i> .	
		<b>NO</b>	No output of values.	
	<b>SCREEN</b>	Output values to screen.		
	<b>YES</b>	Output values to screen and log file.		

Command	PLOT (continued)		
Parameter	Default	Function	
GRAPH	YES	Graph drawing switch:	
		NO YES	Do not draw graph. Draw graph.
TITLE	none	Additional title for graph.	
LOCAL	YES	Local coordinate switch:	
		NO YES	Report coordinates with respect to Global Coordinate System. Report coordinates with respect to local coordinate system.
ORDINATE	NUMBER	Ordinate of graph:	
		DISTANCE NUMBER	DISTANCE along line. Point NUMBER.
OFFSET	0	Offset in distance for overlaying graphs.	
STYLE	AUTOMATIC	Line style:	
		AUTOMATIC	Program chooses a different style for each graph drawn on the same axes.
		0 >0	Solid line. Broken line.
COLOUR	AUTOMATIC	Line colour:	
		AUTOMATIC >0	Program chooses a different colour for each graph drawn on the same axes. Colour number.

### Notes

The **PLOT** command is used to plot graphs and list values of field components previously calculated by the **ARC**, **CIRCLE** or **LINE** commands (page 5-24, page 5-33 and page 5-65). These can be values calculated by the last **ARC**, **CIRCLE** or **LINE** command (**FILE=TEMP**) or read in from a table file by the **PLOT** command. The structure of the files is given in section “**TABLE Files**” on page 5-18. If no file name extension is given the extension *table* is assumed.

The coordinates of the field points can be displayed with respect to the local coordinate system of the **SET** command (**PLOT +LOCAL**) or with respect to the Global Coordinate System (**PLOT -LOCAL**).

The parameters allow choices of where the numeric values should be displayed (**PRINT** options), whether or not a graph should be plotted or not (**+GRAPH** or **-GRAPH**), whether or not a graph should overlay a previously plotted graph (**+ERASE** or **-ERASE**) and what the Y axis scale limits should be (**YMIN** and **YMAX**).

A **TITLE**, in addition to the value of the integral and the **COMPONENT**, can be added to the key at the bottom of the graph.

When multiple graphs are drawn on the same axes, each line is drawn in a different style or colour. The choice of **STYLE** and **COLOUR** can be **AUTOMATIC** or selected by the user.

If more than one graph is drawn on the same set of axes (**-ERASE**), the horizontal coordinates or **ORDINATES** of the points used can correspond to the point **NUMBERS** or to the **DISTANCE** along the line. If **ORDINATE=DISTANCE** an **OFFSET** can be added to the distance along the line.

If a file is not used (**FILE=TEMP**), expressions for the **COMPONENT** can use as variables any of the system variables which are currently available (see “**System Variables**” on page 5-10). If a table file is used, the component name and values are read from the 4<sup>th</sup> column of the file.

The **PLOT** command also calculates the integral of the **COMPONENT**, and stores its value in the system variable **INTEGRAL**. The maximum and minimum values of **COMPONENT** are stored in system variables **MAXIMUM** and **MINIMUM**.

## The **POINT** Command

---

**Summary** Calculate fields values at a point.

**Icon**



**Menu Route:** Fields↓  
Fields at a point

**Command Line Parameters:**

Command	<b>POINT</b>	
Parameter	Default	Function
<b>XP</b>	0	X-coordinate of the point.
<b>YP</b>	0	Y-coordinate of the point.
<b>ZP</b>	0	Z-coordinate of the point.
<b>COMPONENT</b>	X	Field component to be printed.

**Notes**

The **POINT** command evaluates field values at a point. The point is specified by parameters **X**, **Y** and **Z** which are with respect to the local coordinate system of the **SET** command ([page 5-85](#)).

All the currently available field quantities described in section “[System Variables](#)” on [page 5-10](#) are calculated and their values can be used in subsequent **\$ CONSTANT** or **\$ PARAMETER** commands. The **POINT** command prints out the coordinates and the value of the **COMPONENT**. Expressions for the **COMPONENT** can use any of the system variables and any user variable.

## The **POLAR** Command

---

### Summary

Calculate fields over a patch specified in  $r\theta z$  coordinates.

### Icon



### Menu Route:

Fields↓  
Fields on a polar patch

### Command Line Parameters:

Command	<b>POLAR</b>	
Parameter	Default	Function
<b>R1</b>	<i>none</i>	R-coordinate of the first corner of the surface.
<b>T1</b>	<i>none</i>	$\theta$ -coordinate of the first corner of the surface.
<b>Z1</b>	<i>none</i>	Z-coordinate of the first corner of the surface.
<b>R2</b>	<i>none</i>	R-coordinate of the second corner of the surface.
<b>T2</b>	<i>none</i>	$\theta$ -coordinate of the second corner of the surface.
<b>Z2</b>	<i>none</i>	Z-coordinate of the second corner of the surface.
<b>R3</b>	<i>none</i>	R-coordinate of the third corner of the surface.
<b>T3</b>	<i>none</i>	$\theta$ -coordinate of the third corner of the surface.
<b>Z3</b>	<i>none</i>	Z-coordinate of the third corner of the surface.
<b>R4</b>	<i>none</i>	R-coordinate of the fourth corner of the surface.
<b>T4</b>	<i>none</i>	$\theta$ -coordinate of the fourth corner of the surface.
<b>Z4</b>	<i>none</i>	Z-coordinate of the fourth corner of the surface.
<b>N1</b>	10	Number of points on sides 1 and 3.
<b>N2</b>	10	Number of points on sides 2 and 4.

### Notes

The **POLAR** command evaluates field quantities on 4-noded surface patches in cylindrical polar coordinates. The results can be displayed by the **MAP** commands. For each field point all the currently available system variables are calculated and stored (see section “[System Variables](#)” on page 5-10).

The patch is specified by its corner points ( $R1, T1, Z1$ ,  $R2, T2, Z2$ ,  $R3, T3, Z3$  and  $R4, T4, Z4$ ) in cylindrical polar coordinates. The  $\theta=0$  plane is the ZX plane.

The positions of the points are defined in the local coordinate system defined with the **SET** command (page 5-85). The field vectors are evaluated in the Global Coordinate System.

The field quantities are evaluated at  $N1*N2$  points. The coordinates of points other than the corners are found by linear interpolation in R,  $\theta$  and Z using a 4-noded isoparametric two-dimensional finite element shape function. The surface therefore can be a circular cylinder, a plane, a cone or even a spiral. To achieve a circular patch, or a sector of a circle it is necessary to put two of the 4 defining points at zero radius, each with the same  $\theta$  coordinate as one of the other two points.

It is possible to store the evaluated field quantities in a file using the **TABLE** command (page 5-97).

## The **PRINT** Command

---

**Summary** Print the current display.

**Icon**



**Menu Route:**

File↓  
Print

**Command Line Parameters:**

Command	<b>PRINT</b>
No parameters	

**Notes**

This command allows the current display to be printed. A dialog of the available printers and their options for printing is opened. The current display can then be printed on the selected printer.

The printers available are determined from the system.

An image can be saved to file or copied to the clipboard using the **PICTURE** command (“[The PICTURE Command](#)” on page 5-72).

## The **SELECT** Command

---

### Summary

Select surface facets or elements to be displayed.

### Icons



Select



Default select and refresh:

```
SELECT ACTION=DEFAULT
SELECT ACTION=SELECT OPTION=SURFACES
THREED OPTION=REFRESH
```



Repeat select and refresh:

```
SELECT ACTION=SELECT
THREED OPTION=REFRESH
```

### Menu Route:

View↓  
Select

### Command Line Parameters:

Command	SELECT		
Parameter	Default	Function	
ACTION	ADD	Create list of labels or select:	
		ADD	Add label to list.
		DEFAULT	Create a list of labels suitable for type of analysis loaded.
		REMOVE	Remove label from list.
		RESET	Empty the list of labels.
		SELECT	Make selection.

Command	<b>SELECT</b> ( <i>continued</i> )		
Parameter	Default	Function	
<b>LABEL</b>	<b>NOTAIR</b>	Label to be added or removed from list.	
		<i>material_name</i>	Material names including <b>AIR</b> or <b>NOTAIR</b>
		<i>potential_type</i>	Potential type: <b>REDUCED</b> , <b>TOTAL</b> or <b>VECTOR</b> .
		<i>element_type</i>	Element type: <b>LINEAR</b> or <b>QUADRATIC</b> .
		<i>bc_name</i>	Boundary condition name.
		<i>user_label</i>	Labels added by the Modeller or pre processor including <b>ALL</b> .
		<i>drive_label</i>	Drive label: all conductors with this drive.
		<i>conductor_number</i>	Conductor number.
<b>OPTION</b>	<b>SURFACE</b>	Type of selection with <b>ACTION=SELECT</b> .	
		<b>ADD</b>	Select surfaces then add a number of <b>LAYERS</b> of elements.
		<b>SURFACES</b>	Select surfaces.
		<b>ELEMENTS</b>	Select elements.
<b>COILS</b>	<b>NO</b>	Conductor selection switch.	
		<b>NO</b>	Conductors selected for geometry only.
		<b>YES</b>	Conductors selected for field display.

Command	<b>SELECT</b> ( <i>continued</i> )		
Parameter	Default	Function	
<b>CUT</b>	<b>NO</b>	Cut plane switch.	
		<b>NO</b>	No <b>CUT</b> plane.
		<b>YES</b>	Select surfaces in the <b>CUT</b> plane.
		<b>FRONT</b>	Select surfaces in front of <b>CUT</b> plane.
		<b>BACK</b>	Select surfaces behind <b>CUT</b> plane.
<b>THETA</b>	0	$\theta$ Euler angle of <b>CUT</b> plane.	
<b>PHI</b>	0	$\phi$ Euler angle of <b>CUT</b> plane.	
<b>ZCUT</b>	0	Z-coordinate of <b>CUT</b> plane in local coordinate system defined by <b>THETA</b> and <b>PHI</b> .	
<b>ZTOLERANCE</b>	1.0E-5	Tolerance on <b>ZCUT</b> for <b>CUT=YES</b> .	
<b>ACCURACY</b>	0	Maximum facet size on conductors.	
		0	Use element sizes.
		>0	Subdivide elements so that largest subdivision of an element is $\leq$ <b>ACCURACY</b>
<b>LAYERS</b>	1	Number of layers added (or removed if <b>LAYERS</b> is negative) with <b>option=add</b> .	

### Notes

The **SELECT** command creates a 'display buffer' containing a selection of the surfaces of the finite element mesh and conductors which are to be drawn with the **THREED** command (page 5-101). This is a two stage process:

- create a list of labels: labels can be
  - added to the list (**ACTION=ADD LABEL=name**)
  - removed from the list (**ACTION=REMOVE LABEL=name**)
  - all removed from the list (**ACTION=RESET**)

- added or removed in groups with additional names:

Additional labels	Meaning
ALL_BOUNDARIES	all surfaces with boundary condition
ALL_CONDUCTORS	all conductors
ALL_ELEMENTS	all element types
ALL_MATERIALS	all material names
ALL_POTENTIALS	all potential types
ALL_SURFACES	all surface labels
ALL_USERSURFACES	all user surface labels
ALL_USERVOLUMES	all user volume labels
ALL_VOLUMES	all volume labels
NOTAIR	all material names except air

- chosen to match the current simulation type (**ACTION=DEFAULT**).

For conductors, the label name can be either the conductor number or the drive label.

2. operate on the list of labels:

- **OPTION=SURFACES** selects three types of element facets:

facets which lie in Modeller surfaces or pre processor facets which have labels given by the **LABEL** parameter.

facets which lie on the surface of Modeller cells or pre processor volumes which have the labels given by the **LABEL** parameter.

**LABEL=ALL** indicates the element facets on the exterior surface of the mesh.

- **OPTION=ELEMENTS** selects surfaces of all the elements which have volume labels given by the **LABEL** parameter. **LABEL=ALL** indicates all elements.
- **OPTION=ADD** selects surfaces and then moves the selected surface outwards (if number of layers is greater than zero) or inwards (if number of layers is negative) by a number of **LAYERS**.

The part of the finite element mesh selected can also be restricted by a cut plane. The **CUT** parameter can be set to **FRONT** or **BACK** to select surfaces of elements in front or behind the plane or to **YES** to select surfaces which lie in the plane. The tolerance on the position of the plane (**ZTOLERANCE**) should be set greater than zero to ensure that element surfaces are found correctly for **CUT=YES** and facets which touch the cut plane from the “wrong side” are eliminated for **CUT=FRONT** and **CUT=BACK**. The cut plane is defined as local XY plane of a

coordinate system given by Euler angles **THETA** and **PHI** and the normal distance to the global origin **ZCUT**.

The **ACCURACY** parameter can be used to subdivide conductor surface facets. If **ACCURACY>0**, the facets are subdivided into smaller facets so that no facet linear dimension is larger than **ACCURACY**. If **ACCURACY=0**, no subdivision takes place.

## The SET Command

---

### Summary

Set options which affect the field calculations: method, point locations, time, etc.

### Menu Route:

Options↓  
 Field calculations  
 Field points  
 AC time  
 Include Jc in coils

### Command Line

#### Parameters:

Command	SET	
Parameter	Default	Function
FIELD	NODAL	Field calculation method:
		NODAL Interpolation of NODAL values. INTEGRAL INTEGRATION of equivalent magnetisation and current sources.
COIL	NODAL	Field calculation method for conductors in REDUCED potential elements (ignored if FIELD= INTEGRAL).
		NODAL Interpolation of NODAL values. INTEGRAL INTEGRATION of current density.
SEARCH	RESIDENT	SEARCH for field points:
		RESIDENT In RESIDENT (loaded) file only.
		ALL In ALL ACTIVE files.
XLOCAL	0	X-coordinate of the origin of the input local coordinate system.
YLOCAL	0	Y-coordinate of the origin of the input local coordinate system.
ZLOCAL	0	Z-coordinate of the origin of the input local coordinate system.
TLOCAL	0	$\theta$ Euler angle of input local coordinate system.
PLOCAL	0	$\phi$ Euler angle of input local coordinate system.

Command	SET (continued)	
Parameter	Default	Function
SLOCAL	0	$\psi$ Euler angle of input local coordinate system.
TIME	0	Time for steady-state ac results: angle in degrees around ac cycle.
JCOIL	NO	Calculate source current density at field points in coils: YES or NO

### Notes

The **SET** command defines parameters which affect the way the field calculation commands operate.

## Field Calculation Methods

The **FIELD** and **COIL** parameters set the method of field calculation used when processing results. The method also depends on the potential type at the field point.

### Nodal Fields

- **total scalar potential** or **vector potential** regions: The total field is calculated by interpolation of the **NODALLY** averaged values.
- **reduced scalar potential** regions: the field is a combination of the field from the conductors and the field from the finite element mesh. The **COIL** parameter can be set to:
  - **NODAL** to request interpolation of nodal values supplied by the analysis programs.
  - **INTEGRAL** to request evaluation of the conductor fields by direct integration of the defined currents.

### Integral Fields

- The magnetic field strength, **H**, is calculated from magnetisation and current density sources (TOSCA (magnetostatics and current flow) and ELEKTRA simulations only). The field points can be anywhere inside or outside the model.

The **FIELD** and **COIL** parameters have no affect when recovering fields from conductor-only problems. In this case the magnetic field strength, **H** is calculated by integration.

## Field Point Searching

It is possible to have several active database files with the OPERA-3d post processor. The **SEARCH** parameter is used to choose the **RESIDENT** file only or **ALL** active files. If **ALL** is selected then for each field point each file will be loaded in turn until the field point is found.

## Local Coordinate System

The field points given with the **ARC**, **CARTESIAN**, **CIRCLE**, **GRID**, **LINE**, **POINT** and **POLAR** commands are all defined with respect to a local coordinate system for field calculations. This coordinate system is set by its origin (**XLOCAL**, **YLOCAL**, **ZLOCAL**) and Euler angles (**TLOCAL**, **PLOCAL**, **SLOCAL**). Details of the definition of Euler angles are given in section “Euler Angles” on page 2-31. The local coordinate system, if it is different from the Global Coordinate System, is shown on the display in orange.

## Steady-state AC Results

Steady state ac results are stored in the post processor as complex numbers. The real and imaginary parts are combined by the program to provide the field values at an angle around the ac cycle (**TIME=angle**) (see also section “System Variables” on page 5-10).

## Current Density in Coils

For field points inside coils the conductors, the current density vector can optionally be calculated and stored in system variables **JCX**, **JCY** and **JCZ**. This can be selected using **SET +JCOIL**.

## The **SHOW** Command

---

### Summary

List details of the active simulations including the local coordinate systems and replications.

### Menu Route:

File↓  
List active databases

### Command Line Parameters:

Command	<b>SHOW</b>	
Parameter	Default	Function
<b>TITLE</b>	<b>NO</b>	Display user titles: <b>YES</b> or <b>NO</b> .
<b>OPTION</b>	<b>ACTIVE</b>	Listing option:
	<b>ACTIVE</b>	All <b>ACTIVE</b> files.
	<b>FULL</b>	Full details of resident simulation.
	<b>LOADED</b>	All simulations in resident file.

### Notes

The **ACTIVATE** and **LOAD** commands ([page 5-21](#) and [page 5-67](#)) allow many simulations to be active, but only one to be loaded. The same simulation can be activated several times with different parameters. Each database can contain several simulations. To keep track of these files and simulations, the **SHOW** command displays information about their contents.

There are 3 options:

- **OPTION=ACTIVE**: this gives a summary about all the active simulations. More information is given about the currently loaded simulation.
- **OPTION=FULL**: this gives complete information about the currently loaded simulation including all the material properties and analysis specific data.
- **OPTION=LOADED**: this lists all the simulations in the file containing the currently loaded simulation.

The information includes:

Database Information	Value of <b>OPTION</b>		
	<b>ACTIVE</b> All active files	<b>FULL</b> Details of loaded simulation	<b>LOADED</b> All simulations in loaded file
index number	•	•	•
file name	•	•	
analysis program	•	•	•
analysis type	•	•	•
simulation number	•	•	•
number of simulations in the file	•	•	
simulation status	•	•	•
unit set		•	
user title	•	•	
file creation date, directory and computer	•	•	
time point (transient) or frequency (steady-state ac)	•	•	•
all time points (transient) or frequencies (steady state ac)		•	
linear or non-linear analysis	•	•	•
analysis program specific data		•	
number and types of conductors and current density values		•	
boundary condition labels		•	
material names	•	•	
material data		•	
numbers of nodes and elements		•	
local coordinate system and replications	•	•	
coordinate limits	•	•	

## The **SURFACE** Command

---

**Summary** Integrate filed quantity over selected surfaces.

**Icon**



**Menu Route:** Integrals↓  
Other surface integrals

**Command Line Parameters:**

Command	<b>SURFACE</b>	
Parameter	Default	Function
<b>TAVERAGE</b>	<b>YES</b>	Time-average switch:
		<b>NO</b> Calculate integrals at time of <b>SET</b> command.
		<b>YES</b> Calculate time-average integrals.
<b>COMPONENT</b>	<b>X</b>	Field component to be integrated.

**Notes**

The **SURFACE** command integrates a field component expression over the surface selected by the **SELECT** command (page 5-80). Expressions for the **COMPONENT** can use as variables any of the system variables listed in section “System Variables” on page 5-10 and any user variables.

The system variable **INTEGRAL** is updated with the value of the integral.

In steady-state alternating current problems, for integrands which are the product of two field quantities, the integral is a function of time with the form

$$F_x = A + B \cos 2\omega t + C \sin 2\omega t \quad (5.9)$$

The values  $B$  and  $C$  have little meaning on their own. The time-average value,  $A$  is the value commonly required. This can be calculated directly using **+TAVERAGE**.

The values of  $B$  and  $C$  can be found by setting the times to 0, 45 and 90 (see “[The SET Command](#)” on page 5-85), to give values of  $F_x$  at each time:  $F_0$ ,  $F_{45}$  and  $F_{90}$  with the **-TAVERAGE** option.

$$\begin{aligned} A &= \frac{F_0 + F_{90}}{2} \\ B &= \frac{F_0 - F_{90}}{2} \\ C &= F_{45} - A \end{aligned} \tag{5.10}$$

The following commands can be used to achieve this:

- Example - time-average  $F_x$ :

```
set time=0
inte -tave
$ cons #fx0 fx
set time=45
inte
$ cons #fx45 fx
set time=90
inte
$ cons #fx90 fx
$ para #fxa 0.5*(#fx0+#fx90)
$ para #fxb 0.5*(#fx0-#fx90)
$ para #fxc #fx45-#fxa
```

The values  $F_0$ ,  $F_{45}$  and  $F_{90}$  have little meaning on their own.

## The **SYSVARIABLE** Command

---

### Summary

Control which system variables are available for field calculations.

### Menu Route:

Options↓

Add system variables  
Delete system variables  
List variables in database  
List variables in program

### Command Line

#### Parameters:

Command	SYSVARIABLE		
Parameter	Default	Function	
<b>MODE</b>	<b>PROGRAM</b>	Action required:	
		<b>ADD</b>	Add a system variable to the program from the database.
		<b>DBASE</b>	List the system variables in the database.
		<b>DELETE</b>	Delete a system variable from the program.
<b>PROGRAM</b>		List the system variables in the program.	
<b>NAME</b>	<i>none</i>	Stem name of the variable to be added or deleted. ! for a list.	
<b>TYPE</b>	<b>SCALAR</b>	Type of variable to be added or deleted:	
		<b>SCALAR</b>	scalar.
		<b>VECTOR</b>	vector (X, Y and Z components).
<b>UNIT</b>	<b>DEFAULT</b>	Unit conversion expression.	
		<b>DEFAULT</b>	default unit for known variable names.
		<i>expression</i>	unit expression.

Command	SYSVARIABLE (continued)		
Parameter	Default	Function	
REFLECTION	YES	Reflection in model boundaries:	
		DEFAULT	default reflection for known variable names and simulation types.
		INVERSE NO YES	change of sign. no reflection. same sign.

### Notes

The **SYSVARIABLE** command lists, adds and deletes system variables.

Most loading of system variables happens automatically within the program, variables being loaded when they are required. For example, for TOSCA magneto-statics, the potential, field strength and flux density are loaded when a database is activated and loaded. The source field strength is loaded automatically if integral coil fields are selected in the **SET** command. System variables are reloaded when a new simulation is loaded.

However, the user might sometimes need a variable which is available in the database but has not yet been loaded, or, for reasons of efficiency, might want to delete some of the system variables which have been loaded. The **SYSVARIABLE** command provides this functionality as well as listing the variables in the database and program.

See “Current Density in Coils” on page 5-87 for how to add system variables **JCX**, **JCY** and **JCZ**.

## Adding System Variables

When system variables are loaded the program follows this procedure:

- **TYPE=SCALAR**, for example the magnetic scalar potential, **NAME=POT**: The program reads the real part (**RPOT**). If the imaginary part (**IPOT**) is also available it will be read as well. One of the following expressions will be defined:

- Real scalars:

```
POT=RPOT
```

- Complex scalars:

```
POT=RPOT*COST+IPOT*SINT
POTO=SQRT (RPOT**2+IPOT**2)
POTP=ATAN2 (IPOT;RPOT)
```

where **COST** and **SINT** are calculated from the value of AC time defined by the **SET** command (page 5-85).

The following variables can be used by the user: **POT**, **RPOT** and (for steady state ac analyses) **IPOT**, **POT0** and **POTP**.

- **TYPE=VECTOR**, for example the magnetic field strength, **NAME=H**: The program reads each component (real and imaginary parts if available) and defines expressions such as:

– Real vectors:

```
HX=RHX
```

– Complex vectors:

```
HX=RHX*COST+IHX*SINT
HX0=SQRT (RHX**2+IHX**2)
HXP=ATAN2 ( IHX ;RHX)
```

and similarly for the Y and Z components. It also defines

```
HMOD=SQRT (HX**2+HY**2+HZ**2)
```

to give the magnitude of the field strength.

The following variables can be used by the user: **HX**, **RHX**, **HY**, **RHY**, **HZ**, **RHZ**, **HMOD**, and (for steady-state ac analyses) **IHX**, **IHY**, **IHZ**, **HX0**, **HY0**, **HZ0**, **HXP**, **HYP** and **HZP**.

- **TYPE=VECTOR**, for example, **NAME=E\_**: The program interpolates the edge or face values (indicated by ‘\_’) to provide vector quantities, e.g **E\_X**, **E\_Y** and **E\_Z** and **E\_MOD**, at each field point.

The following variables can be used by the user: **E\_X**, **RE\_X**, **E\_Y**, **RE\_Y**, **E\_Z**, **RE\_Z**, **E\_MOD**.

The unit conversion expression for the quantity represented by the system variable should also be given so that the values are scaled appropriately to the user’s choice of units. Unit expressions should be given in terms of the following names, which correspond to the parameter names of the **UNITS** command (page 5-107). The program recognises the variable names used by CARMEN, ELEKTRA, SCALA, SOPRANO and TOSCA and applies the correct unit expression if **UNIT=DEFAULT** is specified.

Unit conversion factors: **LENGU**, **FLUXU**, **FIELU**, **SCALU**, **VECTU**, **CONDU**, **CURDU**, **POWEU**, **FORCU**, **ENERU**, **ELECU**, **DISPU**.

The program also needs to know how to adjust the signs of the system variables when returning values in reflected images of the mesh. **REFLECTION=YES** gives the same symmetry as the principal field of the simulation, i.e. **H** for magnetic field problems (CARMEN, ELEKTRA, SOPRANO and TOSCA) and **E** for electric field problems (SCALA and TOSCA). The program recognises the vari-

able names used by CARMEN, ELEKTRA, SCALA, SOPRANO and TOSCA and applies the correct reflection if **REFLECTION=DEFAULT** is specified.

Remember that when referencing system variables in the **SYSVARIABLE** command, only the stem of the name should be given after removing the leading **R** or **I** and, for a vector, the trailing **X**, **Y** or **Z**.

## Deleting System Variables

When system variables are deleted from the program it follows this procedure:

- **TYPE=SCALAR**, for example the magnetic scalar potential, **NAME=POT**: The program deletes the real part (**RPOT**). If the imaginary part (**IPOT**) is available it will be deleted as well. The following variables will be marked as no longer being available: **RPOT**, **IPOT**.
- **TYPE=VECTOR**, for example the magnetic field strength, **NAME=H**: The program deletes each component (real and imaginary parts if available). The following variables will be marked as no longer being available: **RHX**, **IHX**, **RHY**, **IHY**, **RHZ**, **IHZ**.

Remember that when referencing system variables, only the stem of the name should be given after removing the leading **R** or **I** and, for a vector, the trailing **X**, **Y** or **Z**.

## Listing System Variables

4 types of list are available:

- **MODE=ADD, NAME=!** and **MODE=DBASE** lists the system variables available in the database; **MODE=DBASE** also shows the type of variable:
  - Node No. indexed indicates a variable which is continuous throughout the model.
  - Node indexed multivalued by material indicates a variable which is discontinuous at the surfaces of materials.
  - Node indexed multivalued by potential code indicates a variable which is discontinuous at the surfaces of different potential types.
  - Edge No. indexed indicates solution of edge variable elements. This should be loaded as a vector.
  - Face No. indexed indicates an additional solution from edge variable elements. It should also be loaded as a vector.

In general the last entries in the list are the results of the simulation. The earlier entries are stored by the software to enable restarts.

- **MODE=DELE, NAME=!** and **MODE=PROGRAM** lists the system variables in the program. **MODE=PROGRAM** also shows those defined within the software which are marked (**program**) and are always available and so cannot be deleted. Those marked (**user**) have been loaded from the database and therefore can be deleted.

## The **TABLE** Command

---

**Summary** Read and write table files of field values at points.

**Menu Route:** Tables↓  
Read and write table files

**Command Line Parameters:**

Command	<b>TABLE</b>	
Parameter	Default	Function
<b>INFILE</b>	<i>none</i>	Name of input data file.
<b>OUTFILE</b>	<i>none</i>	Name of output data file.
<b>F1</b>	<b>X</b>	Expression for first column in output data file.
<b>F2</b>	<b>Y</b>	Expression for second column in output data file.
<b>F3</b>	<b>Z</b>	Expression for third column in output data file.
<b>F4</b>	<i>none</i>	Expression for fourth column in output data file.
<b>F5</b>	<i>none</i>	Expression for fifth column in output data file.
<b>F6</b>	<i>none</i>	Expression for sixth column in output data file.
<b>F7</b>	<i>none</i>	Expression for seventh column in output data file.
<b>F8</b>	<i>none</i>	Expression for eighth column in output data file.
<b>F9</b>	<i>none</i>	Expression for ninth column in output data file.
<b>F10</b>	<i>none</i>	Expression for tenth column in output data file.
<b>F11</b>	<i>none</i>	Expression for eleventh column in output data file.
<b>F12</b>	<i>none</i>	Expression for twelfth column in output data file.

**Notes** The **TABLE** command is provided to facilitate an interface to other post processing programs. It performs three tasks:

1. Read an input source, given by **INFILE**, to obtain field point coordinates and other columns of values.
2. Calculate field values at the points, if necessary.
3. Output the fields.

The value of **INFILE** specifies the source of the field point coordinates:

- **INFILE=TEMP**: the field points contained in the internal buffer created by the **ARC**, **CARTESIAN**, **CIRCLE**, **LINE** and **POLAR** commands are processed.
- **INFILE=DBAS**: the coordinates of the nodes in the current database are used.
- **INFILE=SELE**: the coordinates of the nodes on the selected surfaces are used.
- **INFILE=filename**: field point coordinates are read from a table file.

The value of **OUTFILE** specifies the destination of the field values:

- **OUTFILE=TEMP**: the current set of field values is calculated and remain in the programs internal buffer.
- **OUTFILE=DBAS**: this is intended for use with **INFILE=filename**, where the file contains nodal coordinates and additional fields. If the nodal coordinates match the nodes in the current database, the field values will be added to the database.
- **OUTFILE=NULL**: this option does no field calculation or output, but allows the format of an input file to be tested.
- **OUTFILE=filename**: the available set of field values is calculated and the values up to 12 component expressions are output to a table file. The expressions, **F1** to **F12**, can use as variables any of the system variables listed in section “**System Variables**” on page 5-10 and any user variables. They are evaluated at each field point in the input file. All the values for one point appear in one record of the output file.

If no file name extension is given, the extension *table* is supplied for the input and output data files.

The format of the data files is given in section “**TABLE Files**” on page 5-18 which includes an example output data file.

The following sequence of commands can be used to add fields from a magneto-statics analysis to a SCALA database, so that the SCALA particle tracking can make use of the magnetic fields. After these commands have been completed, the SCALA problem should be run in restart mode.

```

/ Activate and load SCALA database
acti file=space
load
/ Form table of node coordinates
tabl infi=dbas outf=nodes
/ Activate and load magnetostatic problem
acti file=magnet
load
/ Form table of magnetic fields at nodes of SCALA

```

```
/ problem
tabl infi=nodes outf=fields f1=x f2=y f3=z f4=rh f5=rhy,
f6=rhz
/ Add the fields to the SCALA database
load file=1
tabl infi=fields outf=dbas
```

This technique can also be used to add magnetic fields to a TOSCA database which contains an electrostatic model or electric fields to a database containing a magnetostatic model, so that combined electric and magnetic field particle tracking can be done (see [“The COMBINE Command” on page 5-37](#)).

## The **TITLE** Command

---

### Summary

Add title, date and time to the display.

### Menu Route

View↓  
Title

### Command Line Parameters

Command	<b>TITLE</b>		
Parameter	Default	Function	
<b>STRING</b>	<i>none</i>	A graphics window title.	
<b>POSITION</b>	<b>TOPLEFT</b>	Title position:	
		<b>BOTTOMCENTRE</b>	Bottom centre
		<b>BOTTOMLEFT</b>	Bottom left
		<b>BOTTOMRIGHT</b>	Bottom right
		<b>NONE</b>	No title
		<b>TOPCENTRE</b>	Top centre
		<b>TOPLEFT</b>	Top left
		<b>TOPRIGHT</b>	Top right
<b>DATE</b>	<b>TOPLEFT</b>	Time/date position:	
		<b>BOTTOMCENTRE</b>	Bottom centre
		<b>BOTTOMLEFT</b>	Bottom left
		<b>BOTTOMRIGHT</b>	Bottom right
		<b>NONE</b>	No date and time
		<b>TOPCENTRE</b>	Top centre
		<b>TOPLEFT</b>	Top left
		<b>TOPRIGHT</b>	Top right

### Notes

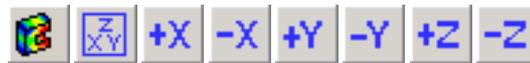
The **TITLE** command controls the display of a title and the date and time. There is a choice of 6 positions for each. If the same position is chosen for both the title and the date, the title appears above the date and time.

# The **THREED** Command

---

**Summary** Control the display of the geometry.

**Icons**



**Menu Routes:**

```
View↓
  3d display
  Refresh
  Views
Fields↓
  Fields on the surface of the model
```

**Command Line Parameters:**

Command	<b>THREED</b>			
Parameter	Default	Function		
<b>TYPE</b>	<b>SURFACE</b>	Type of display:		
		<table border="0"> <tr> <td><b>SURFACE</b></td> <td>Surfaces of the model in material colours.</td> </tr> <tr> <td><b>COMPONENT</b></td> <td>Contours of field <b>COMPONENT</b> on model.</td> </tr> </table>	<b>SURFACE</b>	Surfaces of the model in material colours.
<b>SURFACE</b>	Surfaces of the model in material colours.			
<b>COMPONENT</b>	Contours of field <b>COMPONENT</b> on model.			
<b>COMPONENT</b>	<b>X</b>	Field component for contours.		
<b>MIN</b>	*	Minimum contour value. * for automatic setting.		
<b>MAX</b>	*	Maximum contour value. * for automatic setting.		
<b>VECTORS</b>	<b>NO</b>	Vector display switch:		
		<table border="0"> <tr> <td><b>NO</b></td> <td>Vectors not displayed.</td> </tr> <tr> <td><b>YES</b></td> <td><b>VECTORS</b> displayed as cones at centres of element facets using components <b>VX</b>, <b>VY</b> and <b>VZ</b>, and current directions in conductors.</td> </tr> </table>	<b>NO</b>	Vectors not displayed.
<b>NO</b>	Vectors not displayed.			
<b>YES</b>	<b>VECTORS</b> displayed as cones at centres of element facets using components <b>VX</b> , <b>VY</b> and <b>VZ</b> , and current directions in conductors.			
<b>VX</b>	<b>X</b>	Expression for x-component of vectors.		

Command	<b>THREED</b> (continued)		
Parameter	Default	Function	
<b>VY</b>	<b>Y</b>	Expression for y-component of vectors.	
<b>VZ</b>	<b>Z</b>	Expression for z-component of vectors.	
<b>OPTION</b>	<b>REFRESH</b>	Command option:	
		<b>GETVIEW</b>	Retrieve view parameters after mouse interaction.
		<b>REFRESH</b>	Refresh picture without changing the view.
<b>SETVIEW</b>		Refresh picture using the view parameters.	
<b>SIZE</b>	15	Display extends from the origin by <b>SIZE</b> in each direction. <b>SIZE=0</b> requests the initial view of the model	
<b>ROTX</b>	20	Rotation of model around X axis.	
<b>ROTY</b>	20	Rotation of model around Y axis.	
<b>ROTZ</b>	0	Rotation of model around Z axis.	
<b>XORIGIN</b>	0	X coordinate at centre of picture	
<b>YORIGIN</b>	0	Y coordinate at centre of picture	
<b>ZORIGIN</b>	0	Z coordinate at centre of picture	
<b>PERSPECTIVE</b>	<b>YES</b>	Perspective switch:	
		<b>YES</b>	Perspective view.
		<b>NO</b>	Orthographic view.

### Notes

The **THREED** command updates the 3d picture of the model. The pictures consists of the three dimensional geometry of the finite element mesh and conductors. The **THREED** command uses as data the 'display buffer', which is created by the **SELECT** command. If 'flux-tubes' have been added to the display buffer by the **VIEW** command, they will be displayed as well. Pictures can be line-drawings or can show coloured surfaces. It is also possible to overlay the geometry with displays of the field quantities.

The colours in the pictures can represent the materials (**TYPE=SURFACE**) or alternatively, scalar field quantities, or single components of vector field quantities can be displayed as coloured zone contours (**TYPE=COMPONENT**). **VECTOR** field quantities can be displayed as vectors at the centroid of the element faces.

Expressions for the scalar **COMPONENT** and vector components (**VX**, **VY**, **VZ**) can use as variables any of the system variables listed in section “[System Variables](#)” on page 5-10, any user variable or numerical parameter name.

The values of the maximum and minimum contour zones can be set with **MAX** and **MIN**. If either **MAX** or **MIN** is set to \* the value is calculated from the range of values to be displayed. The number of coloured zones is set by the number of colours available on a particular display type. The system variables **MAXIMUM** and **MINIMUM** are updated by the command.

After a new database is **LOAD**ed, the first use of the **THREEED** command will refresh the display to the default view which matches the size and position of the model. The view can be adjusted using the mouse buttons:

- left button: rotate
- middle button: zoom
- right button: pan

See “[The MOUSE Command](#)” on page 5-71.

The **OPTION** parameter also controls the view of the model:

- **OPTION=GETVIEW**: updates the values of the parameters **SIZE**, **ROTX**, **ROTY**, **ROTZ**, **XORIGIN**, **YORIGIN** and **ZORIGIN**.
- **OPTION=SETVIEW**: uses the current values of **SIZE**, **ROTX**, **ROTY**, **ROTZ**, **XORIGIN**, **YORIGIN** and **ZORIGIN**.
- **OPTION=REFRESH**: updates the picture if necessary without changing the view unless a new database has been loaded in which case the default view is used.

The **MAP** command ([page 5-68](#)) and **VIEW** command ([page 5-110](#)) can add contour maps or particle trajectories to the picture and the different parts of the display (axes, solid colours, outline, contour map, vectors, etc.) can be temporarily removed using the **WINDOW** command ([page 5-119](#)).

## The TRACK Command

---

### Summary

Calculate trajectories of charged particles in electric and magnetic fields.

### Menu Route:

Trajectories↓  
Calculate particle trajectories

### Command Line Parameters:

Command	TRACK	
Parameter	Default	Function
X0	0	Initial X-coordinate of the particle.
Y0	0	Initial Y-coordinate of the particle.
Z0	0	Initial Z-coordinate of the particle.
THETA	0	Theta Euler angle defining particle direction.
PHI	0	Phi Euler angle defining particle direction.
PSI	0	Psi Euler angle defining particle direction.
VOLTAGE	1	Acceleration <b>VOLTAGE</b> or initial energy.
CHARGE	-1	<b>CHARGE</b> on the particle in elementary charge units. -1 is the charge on an electron.
MASS	1	Particle rest <b>MASS</b> in electron rest mass units.
BEAMCURRENT	1	Current associated with the track. It can be an expression in terms of the electric field at the start point.
STEP	1	<b>STEP</b> length between output points.
NSTEP	100	Number of steps to be calculated.
TOLERANCE	0.01	Accuracy required.
OPTIONS	BEAM	Options:
	BEAM	Single <b>BEAM</b>
	FLUX	Flux line
	TEST	A <b>TEST</b> pattern defined by <b>PATTERN</b> and <b>LINE</b> .
PATTERN	1.E-4	Size of the rectangular grid <b>TEST</b> pattern.
LINES	5	Number of X and Y lines in a <b>TEST</b> pattern.
FILE	none	Name of file to save trajectories.

Command	TRACK (continued)		
Parameter	Default	Function	
STATUS	NEW	STATUS of the TRACK file:	
		NEW	Create a NEW file.
		OLD	Append data to an OLD file.
		CLEAR	Overwrite data in an OLD file.

### Notes

The **TRACK** command calculates the trajectories of charged particles through the electric and/or magnetic fields (including full relativistic correction) or follows flux lines. Combined field particle tracking is available for SOPRANO-SS simulations and also when additional fields have been added to an electrostatic or magnetostatics simulation (see section “The **COMBINE** Command” on page 5-37).

The parameter, **OPTION** selects single particles (**OPTION=BEAM**), test patterns (**OPTION=TEST**) or flux lines (**OPTION=FLUX**).

Tracking starts at the point **X0, Y0, Z0**. The initial coordinates (and direction) are defined in the local coordinate system of the **SET** command. In steady-state ac simulations the time at the start of the track is also given by the **SET** command (page 5-85).

For charged particles, the initial direction (defined by the Euler angles **THETA**, **PHI** and **PSI**, see section “Euler Angles” on page 2-31), the initial energy (**VOLTAGE**), **CHARGE** and **MASS** of the particles can be set. The initial direction is along the z-axis of the local coordinate system defined by the Euler angles. The **TEST** pattern option allows many beams of particles to be fired from the plane normal to the initial particle direction, from the intersection points of a square orthogonal grid. The grid can be given a size (**PATTERN**) and the number of grid **LINES** in each direction can be set.

A current is associated with each track. This can be set explicitly to a value in amps or can be calculated by the program from an expression in terms of the electric field (**EX, EY, EZ**), coordinates (**X, Y, Z**) and velocity (**VELX, VELY, VELZ**) at the start of the trajectory.

The calculation is controlled by the **STEP** length along the trajectory, the number of steps (**NSTEP**) and the relative **TOLERANCE** which applies to coordinates and velocities or flux densities. **STEP** only determines the distance between the displayed points on the trajectory, the **TOLERANCE** is achieved by adaptive integration. **STEP** is measured along the trajectory and hence the total trajectory length calculated is **STEP\*NSTEP**.

**Note:** there is a limit of 5000 steps per trajectory.

The trajectory coordinates are stored in a binary file. When the file status is set to **NEW**, a new file will be created to contain the data. Setting status to **OLD** causes the trajectories to be appended to an existing file. The third option **CLEAR** causes an old file to be overwritten. The format of the track file is described in section [page 5-20](#).

The trajectories are drawn on the current display as they are calculated. Subsequently the contents of the **TRACK** files can be **VIEWed** in a number of ways ([page 5-110](#)).

## The UNITS Command

---

**Summary** Set units to be used for physical quantities.

**Icon**



**Menu Route:** Options↓  
Units

**Command Line Parameters:**

Command	UNITS		
Parameter	Default	Function	
LENGTH	CM	Unit for length:	
		CM	centimetre
		INCH	inch
		METRE	metre
		MICRON	micron
	MM	millimetre	
FLUX	GAUSS	Unit for magnetic flux density:	
		GAUSS	gauss
		KGAUSS	kilogauss
	TESLA	tesla	
FIELD	OERSTED	Unit for magnetic field strength:	
		AM	ampere metre <sup>-1</sup>
	OERSTED	oersted	
SCALAR	OCM	Unit for magnetic scalar potential:	
		AMP	ampere
	OCM	oersted centimetre	
VECTOR	GCM	Unit for vector potential:	
		GCM	gauss centimetre
	WBM	weber metre <sup>-1</sup>	

Command	UNITS <i>(continued)</i>		
Parameter	Default	Function	
CONDUCTIVITY	SCM	Unit for conductivity:	
		SCM	siemen centimetre <sup>-1</sup>
		SIN	siemen inch <sup>-1</sup>
		SM	siemen metre <sup>-1</sup>
		SMM	siemen millimetre <sup>-1</sup>
		SMU	siemen micron <sup>-1</sup>
CURD	ACM2	Unit for current density:	
		ACM2	ampere centimetre <sup>-2</sup>
		AIN2	ampere inch <sup>-2</sup>
		AM2	ampere metre <sup>-2</sup>
		AMM2	ampere millimetre <sup>-2</sup>
		AMU2	ampere micron <sup>-2</sup>
POWER	WAT	Unit for power:	
		ERGS	erg second
		HP	horse power
		WATT	watt
FORCE	NEWTON	Unit for force:	
		DYNE	dyne
		GRAMME	gramme force
		KG	kilogramme force
		LBF	pound force
		NEWTON	newton
ENERGY	JOULE	Unit for energy:	
		BTU	British thermal unit
		ERG	erg
		JOULE	joule
ELECTRIC	VCM	Unit for electric field strength:	
		VCM	volt cm <sup>-1</sup>
		VIN	volt inch <sup>-1</sup>
		VM	volt metre <sup>-1</sup>
		VMM	volt millimetre <sup>-1</sup>
		VMU	volt micron <sup>-1</sup>

Command	UNITS (continued)		
Parameter	Default	Function	
DISPLACEMENT	CCM2	Unit for electric flux density:	
		CCM2	coulomb cm <sup>-2</sup>
		CM2	coulomb metre <sup>-2</sup>

### Notes

The **UNITS** command sets the units to be used to interpret user input and display geometric and field data. Each of the parameters can be set to one of a series of predefined character strings corresponding to commonly used units.

**N.B.** The unit of electric scalar potential is always volt and of charge density is coulomb  $length\_unit^{-3}$ .

Unit conversion is performed on the basic system variables. Other system variables are derived from these, and will have the correct units only if consistent units are used for the basic quantities.

When a database is loaded, the current set of units is overwritten with the unit set chosen in the Modeller or pre processor when the database was created (see section “[The SOLVERS Command](#)” on page 4-154).

The current set of units is listed at the right hand side of the display.

The **UNITS** command updates a set of conversion factors for use in expressions. They are based on the equivalent parameters names, taking the first 4 characters, and adding the character **U**, giving: **LENGU**, **FLUXU**, **FIELU**, **SCALU**, **VECTU**, **CONDU**, **CURDU**, **POWEU**, **FORCU**, **ENERU**, **ELECU**, **DISPU**. Values in internal units are divided by the unit factors to convert to user units.

## The **VIEW** Command

---

**Summary**            Display and process particle trajectories.

**Menu Route:**        Trajectories↓  
                           Display particle trajectories  
                           Graph trajectories  
                           Intersect trajectories with patch  
                           Current density map  
                           Create flux tubes

**Command Line  
 Parameters:**

Command	<b>VIEW</b>		
Parameter	Default	Function	
<b>FILE</b>	<i>none</i>	Name of file containing trajectories.	
<b>PLOT</b>	<b>DISPLAY</b>	Type of output required:	
		<b>DISPLAY</b>	<b>DISPLAY</b> trajectories on the current view of the model.
		<b>INTERSECT</b>	Draw graphs of <b>INTERSECTIONS</b> of the trajectories with <b>CARTESIAN</b> or <b>POLAR</b> patch
		<b>TEMP</b>	Calculate current densities from intersections with <b>CARTESIAN</b> or <b>POLAR</b> patch.
		<b>TRACKS</b>	Draw graphs of the trajectories.
<b>TUBE</b>	Create tubes of displayable facets around trajectories.		
<b>XAXIS</b>	<b>Y</b>	The variable plotted on the horizontal axis of the graph.	
<b>YAXIS</b>	<b>X</b>	The variable plotted on the vertical axis of the graph.	
<b>XMIN</b>	*	Lower limit for graph horizontal axis. (* for automatic setting of limit.)	
<b>XMAX</b>	*	Upper limit for graph horizontal axis. (* for automatic setting of limit.)	

Command	VIEW (continued)		
Parameter	Default	Function	
YMIN	*	Lower limit for graph vertical axis. (* for automatic setting of limit.)	
YMAX	*	Upper limit for graph vertical axis. (* for automatic setting of limit.)	
LINE	YES	Graph plotting style:	
		NO	Plot using symbols.
		YES	Plot using lines.
COLOUR	YES	Use of colour in the displays:	
		FUNCTION	Colours represent the values of COMPONENT.
		NO	Use text colour.
		YES	Colours represent trajectory numbers.
COMPONENT	none	Expression used to assign colours to the trajectories when COLOUR=FUNCTION or the radius of the tubes when PLOT=TUBE. The value is also printed when printing is enabled.	
PRINT	NO	Printing switch:	
		NO	No printing.
		YES	All points or intersections printed to dialogue file.
SAMPLE	8	Sample size used to calculate current density for PLOT=INTERSECTIONS and PLOT=TEMP.	
ERASE	YES	Erase previous display (only for PLOT=INTERSECTIONS and PLOT=TRACKS)	
		NO	Add to existing display.
		YES	Erase display first.
SYMMETRY	NO	Include symmetry copies of tracks in DISPLAY, INTERSECTIONS and TUBES: YES or NO.	

### Notes

The VIEW command re-displays and processes trajectories calculated by the TRACK command (page 5-104), or the Space Charge analysis program, and stored in TRACK files (see page 5-20).

## Types of VIEW

- **PLOT=DISPLAY:** this option allows the trajectories to be displayed on the current view of the model using the text colour or multiple colours (page 5-113). The trajectories are overlaid on the three dimensional pictures of the geometry created by the **THREED** command (page 5-101). The tracks can be temporarily removed from the display using the **WINDOW** command (page 5-119).
- **PLOT=TRACKS:** this option displays graphs of the trajectories. The variables plotted on the axes of the graphs can be selected from the set of active system variables (page 5-113). For example to display an axisymmetric projection of the results, with Z on the horizontal axis of the graph, use

```
VIEW XAXIS=Z, YAXIS=R
```

By default, the axis limits are automatically set to contain the functions plotted but limits can be specified.

- **PLOT=INTERSECTIONS:** The intersection positions on the current **CARTESIAN** or **POLAR** patch are calculated for all trajectories and symmetry images. These intersections are ordered with respect to the independent variable defined as the horizontal axis of the output graph. The current in the beams is projected onto the independent variable and the linear current density function, **DENSITY**, is calculated and added to the set of active system variables. The intersections are plotted as graphs using the set of active system variables (page 5-113). When the intersections are calculated, the linear current density (**DENSITY**) is also computed for each intersection point. This is evaluated statistically using varying sample sizes up to a maximum given by the **SAMPLE** parameter. The expected error is also calculated and will be displayed using error bars if **YAXIS=DENSITY** is selected. The sample size should be less than  $\frac{1}{10}$  of the number of intersections. If larger values are used the results will be smooth but the errors will be larger.

For example: to find the radial distribution of current density for a circular beam, use

```
view plot=inte,yaxi=density,
xaxi=sqrt((x-x0beam)**2+(y-y0beam)**2+(z-z0beam)**2)
```

This will collect together all the intersections at the similar radii in order to calculate the density as a function of radial coordinate.

- **PLOT=TEMP:** The intersection positions on the current **CARTESIAN** or **POLAR** patch are calculated for all trajectories and symmetry images. The current density is then calculated for each point on the patch by weighted integration over the patch and the values are smoothed using a fast fourier transform (FFT) with a **SAMPLE** size which determines the amount of smoothing. The current density (**J**) and power density (**PJ**) can then be displayed using the **MAP** command (page 5-68).

- **PLOT=TUBE**: this options creates additional facets in the display buffer. The facets for tubes around the segments of the trajectories, with the radius determined by the **COMPONENT** expression. The tubes can then be displayed using the **THREED** command (page 5-101).

## Viewing In Colour

In each of the **PLOTTING** options, the lines can be displayed using the text colour (**COLOUR=NO**), with colours representing the trajectory number (**COLOUR=YES**) or with colours representing some characteristic of the position along the trajectory (**COLOUR=FUNCTION**).

Functional colours are selected using the value of the **COMPONENT** expression. **COMPONENT** can be assigned to expressions in terms of the active system variables listed below. For example,

**COLOUR=FUNCTION, COMPONENT=TOF**

will display the graphs coloured according to the time of flight from the start points.

## Active System Variables

The **VIEW** command has its own set of system variables which are available in expressions for the graph plotting variables (**XAXIS** and **YAXIS**) and the **COMPONENT** used to determine the colours of the lines.

System Variables for <b>PLOT=DISPLAY</b> and <b>PLOT=TRACKS</b>	
<b>X</b>	X coordinates of points on the trajectory.
<b>Y</b>	Y coordinates of points on the trajectory.
<b>Z</b>	Z coordinates of points on the trajectory.
<b>R</b>	Radial coordinate of points on trajectory.
<b>VELX</b>	X component of particle velocity.
<b>VELY</b>	Y component of particle velocity.
<b>VELZ</b>	Z component of particle velocity.
<b>CURRENT</b>	The current in the track (beamlet).
<b>TOF</b>	Time of flight from the start of the trajectory.
<b>Q</b>	Electronic charge on the particle.
<b>M</b>	Mass of the particle in electron units.

<b>System Variables for PLOT=INTERSECTIONS</b>	
<b>X</b>	X coordinates of the intersection points.
<b>Y</b>	Y coordinates of the intersection points.
<b>Z</b>	Z coordinates of the intersection points.
<b>XOBEAM</b>	X coordinate of the centre of the beam on the intersection surface.
<b>YOBEAM</b>	Y coordinate of the centre of the beam on the intersection surface.
<b>ZOBEAM</b>	Z coordinate of the centre of the beam on the intersection surface.
<b>TXBEAM</b>	X component of the mean direction vector for the beam at the intersection surface.
<b>TYBEAM</b>	Y component of the mean direction vector for the beam at the intersection surface.
<b>TZBEAM</b>	Z component of the mean direction vector for the beam at the intersection surface.
<b>DENSITY</b>	The calculated linear current density in the beam defined by the set of tracks in the track file (units are amp).
<b>VELX</b>	X component of particle velocity.
<b>VELY</b>	Y component of particle velocity.
<b>VELZ</b>	Z component of particle velocity.
<b>CURRENT</b>	The current in the track (beamlet).
<b>TOF</b>	Time of flight from the start of the trajectory.
<b>Q</b>	Electronic charge on the particle.
<b>M</b>	Mass of the particle in electron units.

<b>System Variables for PLOT=TUBE</b>	
<b>X</b>	X coordinates of the intersection points.
<b>Y</b>	Y coordinates of the intersection points.
<b>Z</b>	Z coordinates of the intersection points.
<b>VELX</b>	X component of particle velocity.
<b>VELY</b>	Y component of particle velocity.
<b>VELZ</b>	Z component of particle velocity.
<b>CURRENT</b>	The current in the track (beamlet).
<b>TOF</b>	Time of flight from the start of the trajectory.

<b>System Variables for PLOT=TUBE (continued)</b>	
<b>Q</b>	Electronic charge on the particle.
<b>M</b>	Mass of the particle in electron units.
All available field quantities.	

## Printing Trajectory Data

When printing is selected (**PRINT=YES**) and **COLOUR=FUNCTION** has been set, the following data is output to the dialogue file, *Opera3d\_Post\_n.lp*:

- **PLOT=DISPLAY** and **PLOT=TRACKS**, for all points in all trajectories:  
X Y Z VELX VELY VELZ COMPONENT
- **PLOT=INTERSECTIONS**, for all intersections:  
CURRENT X Y Z VELX VELY VELZ COMPONENT

**N.B.** The unit of current density is amp  $length\_unit^2$ .

## The **VOLUME** Command

---

**Summary** Integrate field quantities over volumes.

**Icon**



**Menu Route:** Integrals↓  
Other volume integrals

**Command Line Parameters:**

Command	<b>VOLUME</b>		
Parameter	Default	Function	
<b>ACTION</b>	<b>INTEGRATE</b>	Create list of volumes or integrate:	
		<b>ADD</b>	Add volume(s) to list.
		<b>INTEGRATE</b>	Integrate.
		<b>REMOVE</b>	Remove volume(s) from list.
		<b>RESET</b>	Empty list of volumes.
<b>LABEL</b>	<b>ALL_VOLUMES</b>	Volumes to be added or removed from list:	
		<i>material</i>	Material name.
		<i>label</i>	Volume label.
		<b>ALL_VOLUMES</b>	All volumes.
<b>TAVERAGE</b>	<b>YES</b>	Time-average switch.	
		<b>NO</b>	Calculate integrals at time of <b>SET</b> command.
		<b>YES</b>	Calculate time-average integrals.

Command	VOLUME (continued)			
Parameter	Default	Function		
ADAPTIVE	NO	Adaptive integration switch.		
		<table border="1"> <tr> <td>NO</td> <td>Use 8 gauss-points in each element.</td> </tr> <tr> <td>YES</td> <td>Use up to 216 gauss-points in each element.</td> </tr> </table>	NO	Use 8 gauss-points in each element.
NO	Use 8 gauss-points in each element.			
YES	Use up to 216 gauss-points in each element.			
COMPONENT	X	Field component to be integrated.		

### Notes

The **VOLUME** command integrates a field component expression in the whole problem space or in labelled sets of elements. Expressions for the **COMPONENT** can use as variables any of the system variables listed in section “System Variables” on page 5-10 and any user variables.

The command operates as a 2-stage process:

- form a list of volumes. Initially all volumes are in the list.
  - ACTION=RESET** empties the list.
  - ACTION=ADD** adds volumes to the list by **LABEL**.
  - ACTION=REMOVE** removes volumes from the list by **LABEL**.
  - Labels can be material names, volume labels including element and potential types and user labels or **ALL\_VOLUMES**.
- integrate forces using **ACTION=INTEGRATE**.

The system variable **INTEGRAL** is updated with the value of the integral.

The basis for the integration is the finite element mesh. The integrals are performed in each element using first-order gaussian quadrature. However, in reduced potential volumes if the coil field is calculated by integration or anywhere if the total field is calculated by integration, the first-order quadrature is insufficient to match the field variation in an element. Switching on adaptive integration (**+ADAPTIVE**) enables the program to use up to 9<sup>th</sup>-order gaussian quadrature in each element to increase the accuracy of the integrals. See section “The SET Command” on page 5-85 for information about field calculation methods.

Reflected and rotated images of the model, specified by the **ACTIVATE** command, are NOT included in the integrations done by the **VOLUME** command.

In steady-state alternating current problems, for integrands which are the product of two field quantities, the integral is a function of time with the form

$$E = A + B \cos 2\omega t + C \sin 2\omega t \quad (5.11)$$

The values  $B$  and  $C$  have little meaning on their own. The time-average value,  $A$  is the value commonly required. This can be calculated directly using **+TAVERAGE**.

The values of  $B$  and  $C$  can be found by setting the times to 0, 45 and 90 (see section “[The SET Command](#)” on page 5-85), to give values of  $E$  at each time:  $E_0$ ,  $E_{45}$  and  $E_{90}$  with the **-TAVERAGE** option.

$$\begin{aligned} A &= \frac{E_0 + E_{90}}{2} \\ B &= \frac{E_0 - E_{90}}{2} \\ C &= E_{45} - A \end{aligned} \quad (5.12)$$

The following commands can be used to achieve this:

- Example - time-average energy:

```
set time=0
volu comp=0.5*(bx*hx+by*hy+bz*hz) -tave
$ cons #en0 integral
set time=45
volu
$ cons #en45 integral
set time=90
volu
$ cons #en90 integral
$ para #ena 0.5*(#en0+#en90)
$ para #enb 0.5*(#en0-#en90)
$ para #enc #en45-#ena
```

The values  $E_0$ ,  $E_{45}$  and  $E_{90}$  have little meaning on their own.

## The **WINDOW** Command

**Summary** Show or hide parts of the display.

**Icons**



**Menu Route:** View↓  
Parts of the display

**Command Line Parameters:**

Command	<b>WINDOW</b>	
Parameter	Default	Function
<b>AXES</b>	<b>YES</b>	Show coordinate axes: <b>YES</b> or <b>NO</b>
<b>SOLID</b>	<b>YES</b>	Show solid view of model: <b>YES</b> or <b>NO</b>
<b>OUTLINE</b>	<b>YES</b>	Show outline view of model: <b>YES</b> or <b>NO</b>
<b>VECTORS</b>	<b>YES</b>	Show vectors on the surface of the model: <b>YES</b> or <b>NO</b>
<b>CONTOURMAP</b>	<b>YES</b>	Show contour map: <b>YES</b> or <b>NO</b>
<b>VECTORMAP</b>	<b>YES</b>	Show vector map: <b>YES</b> or <b>NO</b>
<b>TRACKS</b>	<b>YES</b>	Show trajectories: <b>YES</b> or <b>NO</b>
<b>LABELS</b>	<b>YES</b>	Show contour or trajectory labels: <b>YES</b> or <b>NO</b> .

The **WINDOW** command can be used to hide or show again parts of the display which exist. For example, if a contour map is added to the display with the **MAP** command, it can be hidden using **WINDOW CONTOURMAP=NO** or shown again using **WINDOW CONTOURMAP=YES**.



# INDEX

## Symbols

!	2-5
!!	2-14
#	
free format variables	2-16
user variables	2-19
<b>\$ ABORT</b>	2-14
<b>\$ ASK</b>	2-23
<b>\$ ASSIGN</b>	2-26
<b>\$ BACKSPACE</b>	2-26
<b>\$ CD</b>	2-27
<b>\$ CLOSE</b>	2-24
<b>\$ COMINPUT</b>	2-22
<b>\$ CONSTANT</b>	2-19
<b>\$ DO-loops</b>	2-17
<b>\$ ELIF</b>	2-18
<b>\$ ELSE</b>	2-18
<b>\$ END</b>	2-19
<b>\$ EULER</b>	2-31
<b>\$ FOR-each loops</b>	2-18
<b>\$ FORMAT</b>	2-25
<b>\$ IF</b>	2-18
<b>\$ OPEN</b>	2-24
<b>\$ OS</b>	2-27
<b>\$ PARAMETER</b>	2-19
<b>\$ PAUSE</b>	2-23
<b>\$ PROMPT</b>	2-23
<b>\$ READ</b>	2-24
<b>\$ SKIP</b>	2-14
<b>\$ STRING</b>	2-19

<b>\$ WHILE-loops</b>	2-18
<b>\$ WRITE</b>	2-25
<b>\$-commands</b>	2-17
\$constant	3-122
\$parameter	3-122
<b>\$VFDIR</b>	
<i>see</i> <b>VFDIR</b>	
<b>%INT, %REAL, %EXPR</b>	2-11
<b>%VFDIR%</b>	
<i>see</i> <b>VFDIR</b>	
,, comma	2-6
/, comment	2-30
;, semicolon	2-10
, command separator	2-30
', quotation mark	2-11

## Numerics

20-node brick	3-34
3d Viewer menus	4-174
8 node brick	3-32

## -A-

abbreviation	2-4, 2-9
abort	
<b>DEFINE</b> command	4-55, 4-60, 4-65, 4-71
<b>ACTIVATE</b> command	5-21
adaptive integration methods for RHS	4-81
adaptive time-step	3-20
adding labels	3-68

air ..... 3-75, 4-131  
 analysis ..... 3-20, 3-106  
   frequencies ..... 3-48  
**ANALYSISDATA** command ..... 3-18  
 anisotropic materials ..... 4-15, 4-82  
 anisotropy ..... 3-75, 4-131  
 arc ..... 3-23, 3-44  
   conductors ..... 4-36  
**ARC** command  
   modeller ..... 3-22  
   post processor ..... 5-24  
**ARITHMETIC** command ..... 5-26  
 arrow keys ..... 4-3, 4-5  
**ASK**  
   *see* **\$ ASK**  
 aspect ratio searching ..... 4-52  
**ASSIGN**  
   *see* **\$ ASSIGN**  
 automatic potential cuts ..... 3-20, 4-169

### -B-

background ..... 3-95  
**BACKSPACE**  
   *see* **\$ BACKSPACE**  
 backup file ..... 4-112  
 base plane ..... 4-50  
   subdivision editing ..... 4-111  
   subdivision mode ..... 4-68  
 bedstead ..... 3-25, 3-44, 4-30  
**BEDSTEAD** command ..... 3-24  
 BH curves ... 3-75, 4-15, 4-131, 4-165, 5-27  
**BHDATA** command  
   post processor ..... 5-27  
   pre processor ..... 4-15  
 bitmap ..... 3-88, 5-72  
**BLEND** command ..... 3-26  
 block ..... 3-27  
**BLOCK** command ..... 3-27  
*bmp* file format ..... 3-88, 5-72  
**BODY** command ..... 5-28  
 boolean

operations ..... 3-40  
 parameters ..... 2-11  
**BOUNDARY** command ..... 3-28  
 boundary conditions ..... 3-29  
   **DEFINING** ..... 4-87  
   Definition Mode ..... 4-86  
     **EXTENDING** the mesh ..... 4-114  
 derivative ..... 4-89, 4-93  
 displaying ..... 4-101, 4-173  
 functional ..... 4-90  
 incident field ..... 4-89  
 labels ..... 4-159  
 Modification Mode ..... 4-148  
 natural ..... 4-89  
 non-zero total scalar potential 4-79, 4-81  
 normal field ..... 4-87, 4-93  
 perfect conductor ..... 4-88  
 potential ..... 4-88, 4-89, 4-93  
 radiation ..... 4-88, 4-93  
 slip ..... 4-88, 4-89  
 symmetry ..... 4-88, 4-89, 4-93  
 tangential field ..... 4-87, 4-93  
 vector potential ..... 4-88, 4-89, 4-93  
 voltage ..... 4-89

boundary label ..... 3-29, 3-56  
**BOUNDARY** sub-command  
   **IDEAS** command ..... 4-120  
 brick ..... 3-32, 3-34  
   20 node ..... 3-44  
   8 node ..... 3-44  
 brick conductors ..... 4-37, 4-50  
**BRICK20** command ..... 3-33  
**BRICK8** command ..... 3-31  
 built-in commands ..... 2-17

### -C-

*c* ..... 5-15  
 calculator ..... 2-21  
 CARMEN ..... 1-2  
**CARTESIAN** command ..... 5-31  
**CASE** sub-command

- SOLVERS** command ..... 4-161
- CASES** ..... 5-14, 5-22
- CD**
  - see* **\$ CD**
- CDBox ..... 4-3, 4-6
- cell properties ..... 3-35
- CELLDATA** command ..... 3-35
- chamfer ..... 3-26
- character values ..... 2-10
- character variables ..... 2-21
- charge density ..... 3-127, 4-81, 5-12
  - unit ..... 5-109
- CHECK** command
  - modeller ..... 3-37
  - pre processor ..... 4-18
- CHECK** sub-command
  - SOLVERS** command ..... 4-162
- checking ..... 3-37
- CIRCLE** command ..... 5-33
- CLEAR** command
  - modeller ..... 3-38
  - post processor ..... 5-34
  - pre processor ..... 4-21
- clipboard ..... 3-88, 5-72
- CLOSE**
  - see* **\$ CLOSE**
- coenergy ..... 5-47
- coercive field ..... 4-165, 5-12
- coil field integrals ..... 3-20
- COLOUR** command
  - modeller ..... 3-39
  - post processor ..... 5-35
  - pre processor ..... 4-22
- ColourBox ..... 4-3, 4-8
- colours ..... 3-39
- COMBINE** command
  - modeller ..... 3-40
  - post processor ..... 5-37
- combining bodies ..... 3-40
- COMINPUT**
  - see* **\$ COMINPUT**
- command line interface ..... 2-1
- command separator ..... 2-30
- commands ..... 2-4
  - built-in ..... 2-17
  - post processor ..... 5-3
  - pre processor ..... 4-10
- COMMENT** command ..... 3-42
- comments ..... 2-30
- COMPONENT** ..... 2-6, 5-10
- conditional commands ..... 2-18
- conductivity ..... 3-75, 4-131, 4-165, 5-14
  - unit ..... 4-158, 5-107
- CONDUCTOR**
  - DEFINE** sub-command ..... 4-40
  - ERASE** sub-command ..... 4-43
  - MODIFY** sub-command ..... 4-44
  - PRINT** sub-command ..... 4-47
  - QUIT** sub-command ..... 4-48
  - WRITE** sub-command ..... 4-48
- CONDUCTOR** command
  - modeller ..... 3-43
  - post processor ..... 5-39
  - pre processor ..... 4-24
- conductor file ..... 3-55, 3-67, 5-43
- conductor symmetries
  - see* symmetry
  - conductors
- conductors
  - arcs ..... 3-23, 4-36
  - bedsteads ..... 3-25, 4-30
  - bricks ..... 3-32, 4-37
  - constant perimeter ends ..... 3-62, 3-113, 4-34
  - eddy currents ..... 4-80
  - EXPORT** command ..... 3-55
  - forces ..... 5-28
  - helical ends ..... 3-64, 4-32
  - IMPORT** command ..... 3-67
  - in the mesh ..... 4-79, 4-80
  - racetracks ..... 3-93, 4-29
  - single filament ..... 4-25

solenoids ..... 3-104, 4-27  
 straight bars ..... 3-109, 4-35  
 cone ..... 3-47  
**CONSTANT**  
   *see* **\$ CONSTANT**  
 constant perimeter end .... 3-44, 3-62, 3-113,  
   4-34  
 construction lines ..... 4-53  
 continuation lines ..... 2-6  
 contour ..... 3-45, 3-124  
**CONTOUR** command ..... 3-45  
 contour maps ..... 5-69, 5-102, 5-119  
 coordinate system  
   *see* local coordinate system  
 copy ..... 3-119  
 copy picture ..... 3-88, 5-72  
 copying base plane facets ..... 4-65  
 correcting errors ..... 3-121  
 creating objects .... 3-27, 3-47, 3-91, 3-107,  
   3-117  
 cube ..... 3-27  
**CURRENT** ..... 5-113, 5-114  
 current  
   charged beam ..... 5-105  
 current density ..... 3-127, 4-25, 5-12  
   charged beam ..... 5-15  
   unit ..... 4-158, 5-107  
   units ..... 5-115  
   vector ..... 4-82, 4-85  
 current flow ..... 4-81, 4-155, 5-14, 5-86  
 cuts ..... 4-79  
 cylinder ..... 3-47  
**CYLINDER** command ..... 3-47

### -D-

data ..... 3-70  
 data files  
   **\$ COMINPUT** ..... 4-1, 5-2  
   BH curves ..... 4-15  
   command input ..... 2-22  
   conductors ..... 3-55, 3-67, 4-48, 5-43  
   database ..... 4-154, 4-156, 4-157, 4-158,  
     5-18, ..... 5-22, 5-67, 5-88  
   grid ..... 5-18, 5-56  
   *opc* ..... 3-71, 3-96  
   post processor ..... 5-18  
   pre processor 4-109, 4-112, 4-149, 4-179  
   *sat* ..... 3-71, 3-96  
   table .. 5-18, 5-51, 5-56, 5-69, 5-74, 5-97  
   time-tables ..... 4-161  
   **TRACK** ..... 5-20, 5-106  
   universal ..... 4-118  
 data storage level .... 3-35, 3-53, 3-56, 3-125  
 database ..... 3-42, 3-106  
**DBCASEDATA** command ..... 3-48  
 DE curves ..... 4-15  
**DEBUG** ..... 4-18, 4-115  
**DEFINE** command ..... 4-49  
**DEFINE** sub-command  
   **CONDUCTOR** command ..... 4-40  
 defining variables ..... 3-122  
 delete ..... 3-49  
**DELETE** command ..... 3-49  
**DENSITY** ..... 5-114  
 derivative boundary condition  
   *see* boundary conditions  
   derivative  
     **DEVICE** command ..... 4-95  
 device nomination ..... 4-1  
 DialogBox ..... 4-3, 4-7, 4-8  
 dialogue file ..... 2-3  
 discretisation ..... 4-49, 4-51, 4-60, 4-63  
 display ..... 3-45, 3-65, 3-99, 3-124  
   colours ..... 3-39  
**DISPLAY** command ..... 4-97  
**DO**  
   *see* **\$ DO**  
 double-click ..... 4-6  
 draft angle ..... 3-111  
**DRIVE** command ..... 3-50  
 drive label ..... 3-30, 4-25, 4-90, 4-159  
**DRIVE** sub-command

**SOLVERS** command ..... 4-163  
 drives ..... 4-163  
**DUMP** command ..... 4-104

### -E-

$\epsilon_0$  ..... 5-15  
 eddy currents ..... 4-80, 4-155  
   **ELEKTRA** ..... 4-82  
   motion induced ..... 4-82  
 edge properties ..... 3-53  
**EDGEDATA** command ..... 3-53  
**EDIT** command ..... 4-109  
 eigen-values ..... 4-155, 4-161  
 electric field strength ..... 5-12  
   unit ..... 4-158, 5-107  
 electric flux density ..... 5-12  
   unit ..... 4-158, 5-107  
 electric scalar potential ..... 5-12  
   unit ..... 4-158, 5-109  
 electrostatics ..... 4-81, 4-155  
   **SCALA** ..... 4-81  
   **TOSCA** ..... 4-81  
**ELEKTRA** ..... 1-2, 3-20, 4-82, 4-88, 4-155,  
   5-12  
**ELEKTRA-TR** output times ..... 3-48  
 element types ..... 3-35, 3-56, 4-158  
   **DEFINING** ..... 4-81  
   displaying ..... 4-101, 4-173  
**ELIF**  
   *see* **\$ ELIF**  
**ELSE**  
   *see* **\$ ELSE**  
**END** command  
   modeller ..... 3-54  
   post processor ..... 5-45  
   pre processor ..... 4-112  
**END-DO**  
   *see* **\$ END**  
**END-FOR**  
   *see* **\$ END**  
**END-IF**

*see* **\$ END**  
**END-WHILE**  
   *see* **\$ END**  
 energy  
   unit ..... 5-107  
**ENERGY** command ..... 5-46  
 environment variables 2-11, 2-29, 4-7, 4-95  
**ERASE** sub-command  
   **CONDUCTOR** command ..... 4-43  
 Euler angles . 2-31, 3-69, 3-119, 4-26, 4-51,  
   4-82, ..... 5-87  
**EXPORT** command ..... 3-55  
 exporting conductors ..... 3-55  
 exporting data ..... 3-96  
 expressions ..... 2-8  
   boundary conditions ..... 4-90  
   post processing ..... 5-10  
**EXTEND** command ..... 4-113  
 extending volumes ..... 3-111  
**EXTERNAL** ..... 4-18  
 external fields ..... 3-20, 4-169  
 extrusion ..... 3-111, 4-51, 4-71  
   direction ..... 4-51  
   **EXTENDING** the mesh ..... 4-113  
 extrusion direction ..... 4-51

### -F-

**FACEDATA** command ..... 3-56  
 Facet Definition Mode ..... 4-62  
   **EDITING** ..... 4-111  
 Facet Group Operations Mode ..... 4-65  
 field calculation methods ..... 5-86  
 field point coordinates ..... 5-12  
 field values  
   along a line ..... 5-65  
   at a point ..... 5-76  
   on a general surface ..... 5-32, 5-77  
 field vectors ..... 5-69, 5-119  
 filamentary conductors ..... 4-25  
 file  
   open ..... 3-71



- J-**
- Jacobean ..... 4-19
- JE curves ..... 4-15
- joining bodies ..... 3-40
- L-**
- label ..... 3-68
- LABEL** command
- modeller ..... 3-68
- pre processor ..... 4-125
- labels ..... 4-50
- boundary conditions ..... 4-88, 4-159
- DEBUG** ..... 4-18
- displaying ..... 4-101, 4-172
- element types ..... 4-82
- EXTERNAL** ..... 4-18
- material names ..... 4-82
- potential types ..... 4-82
- user defined ..... 4-125
- laminations ..... 4-15, 4-82
- LCS ..... 3-69
- name ..... 3-95
- LCS** command ..... 3-69
- Legendre polynomial fitting ..... 5-51
- length
- unit ..... 4-158, 5-107
- limits ..... 1-3
- LINE** command ..... 5-65
- linear elements ..... 3-106, 4-81
- linear extrusions ..... 4-51, 4-71
- linear motion ..... 4-155
- linearity ..... 3-75, 4-131
- list ..... 3-59, 3-70
- LIST** command ..... 3-70
- LOAD** command
- modeller ..... 3-71
- post processor ..... 5-67
- local coordinate system ..... 3-69, 3-128
- anisotropic materials ..... 4-82
- conductors ..... 4-26
- DEFINE** command (UVW) ..... 4-51
- field calculations ..... 5-75, 5-87
- name ..... 3-95
- permanent magnets ..... 4-82
- post processing ..... 5-22
- system variables ..... 5-14
- loops ..... 2-17
- M-**
- M** ..... 5-113, 5-114, 5-115
- $\mu_0$  ..... 5-15
- magnetic field strength ..... 5-12
- unit ..... 4-158, 5-107
- magnetic flux density ..... 5-12
- unit ..... 4-158, 5-107
- magnetic scalar potential ..... 3-29, 5-12
- unit ..... 4-158, 5-107
- magnetic vector potential ..... 3-29, 5-12
- unit ..... 4-158, 5-107
- magnetostatics ..... 4-79, 4-155
- MAP** command ..... 5-68
- material
- label ..... 3-35, 3-75, 4-131
- name ..... 4-79, 4-158, 4-165
- NULL** ..... 3-79
- orientation ..... 4-85
- properties ..... 3-75, 4-131, 4-164
- Material Definition Mode ..... 4-78
- CARMEN ..... 4-80
- ELEKTRA ..... 4-80
- EXTENDING** the mesh ..... 4-114
- SOPRANO ..... 4-81
- TOSCA ..... 4-79
- Material Modification Mode ..... 4-147
- MATERIAL** sub-command of **I DEAS** ..... 4-122
- MATERIAL** sub-command of the **SOLVERS** command ..... 4-164
- MATERIALS** command
- modeller ..... 3-73
- pre processor ..... 4-129
- maximum value ..... 5-15

Maxwell Stress  
 integral ..... 5-62  
 menu interface ..... 2-1  
 menu off ..... 4-9  
 menus ..... 4-3  
 merging faces ..... 3-26  
**MESH** command  
 modeller ..... 3-77  
 pre processor ..... 4-133  
 mesh control ..... 3-35, 3-56  
 mesh size ..... 3-53, 3-125  
 MessageBox ..... 2-23, 2-30, 4-3, 4-7  
 minimum value ..... 5-15  
**MODEL** command ..... 3-79  
**MODIFY** command ..... 4-135  
**MODIFY** sub-command  
**CONDUCTOR** command ..... 4-44  
**MOUSE** command  
 post processor ..... 3-81, 5-71  
 moving parts ..... 4-82  
 multiply connected ..... 3-20, 4-169  
 multipole moments ..... 5-49  
 multi-valued potential ..... 4-79, 4-81

**-N-**

name ..... 3-95  
 negative periodicity ..... 3-84  
**NODALLY** averaged fields ..... 5-86  
 non-linear ..... 3-75, 4-131  
 non-linear analysis ..... 3-20, 4-169  
 non-linear analysis options ..... 3-20  
 normal electric ..... 3-29  
 normal field boundary condition  
 see boundary conditions  
 normal field  
 normal magnetic ..... 3-29  
 normal unit vector ..... 5-12  
**NOW** ..... 2-21  
**NULL** material ..... 3-79  
 numeric values ..... 2-8  
 numerical variables ..... 2-19

**-O-****OPEN**see **\$ OPEN**

OPERA-3d ..... 1-1  
*Opera3d\_Modeller\_n.log* ..... 2-3  
*Opera3d\_Modeller\_n.lp* ..... 2-3  
*Opera3d\_Post\_n.log* ..... 2-3  
*Opera3d\_Post\_n.lp* ..... 2-3, 5-115  
*Opera3d\_Pre\_n.log* ..... 2-3  
*Opera3d\_Pre\_n.lp* ..... 2-3  
 operaanl ..... 2-28  
 operating system commands ..... 2-27  
 orientation ..... 3-127  
**OS**  
 see **\$ OS**  
 output times ..... 3-48

**-P-**

$\pi$  ..... 5-15  
 packing factor ..... 3-127, 4-82  
**PARAMETER**  
 see **\$ PARAMETER**  
 ParameterBox ..... 4-3, 4-4, 4-8  
 parameters ..... 2-4  
 particle trajectories ..... 5-111, 5-119  
 paste ..... 3-88, 5-72  
**PAUSE**  
 see **\$ PAUSE**  
 PEC ..... 3-29  
 ELEKTRA ..... 4-88  
 SOPRANO ..... 4-88  
 perfect conductor boundary condition  
 see boundary conditions  
 perfect conductor  
 periodic boundary ..... 3-82  
 periodicity ..... 3-20, 3-82, 4-169  
 periodicity boundary condition  
 see boundary conditions  
 symmetry  
**PERIODICITY** command ..... 3-82  
**PERIODICITY** sub-command

- SOLVERS** command ..... 4-165
  - permanent magnets ..... 4-82
  - permeability ..... 3-75, 4-131, 4-165, 5-14
  - permittivity ..... 3-75, 4-131, 4-165, 5-14
  - phase angle of conductors ..... 4-25
  - phase lag ..... 3-75, 4-131
  - pick ..... 3-59
  - PICK** command ..... 3-85
  - pick options ..... 3-59
  - picking ..... 4-4
  - picking filter ..... 3-59
  - PICTURE** command
    - modeller ..... 3-88
    - post processor ..... 5-72
  - planar face ..... 3-111
  - PLOT** command ..... 5-73
  - png* file format ..... 3-88, 5-72
  - POINT** command ..... 5-76
  - Point Definition Mode ..... 4-55
    - EDITING** ..... 4-111
  - Point Modification Mode ..... 4-136
  - POLAR** command ..... 5-77
  - positional assignments ..... 2-6
  - positive periodicity ..... 3-84
  - potential boundary condition
    - see* boundary conditions
    - potential
  - potential cut ..... 3-20, 4-79, 4-169
  - potential types ..... 3-35, 4-158
    - DEFINING** ..... 4-79
    - displaying ..... 4-101, 4-173
  - POTENTIAL\_CUTN** ..... 4-79
  - potentials
    - multi-valued ..... 4-79, 4-81
    - potential cut ..... 4-79
    - reduced scalar ..... 4-79, 4-80
    - single valued ..... 4-79, 4-80
    - total scalar ..... 4-79, 4-80, 4-81
    - vector ..... 4-80, 4-81
  - power
    - density ..... 5-15
    - unit ..... 5-107
  - power loss
    - integral ..... 5-47
  - PRECISIONDATA** command ..... 3-89
  - PRINT** command
    - modeller ..... 3-90
    - post processor ..... 5-79
  - PRINT** sub-command
    - CONDUCTOR** command ..... 4-47
  - printing ..... 3-90, 5-79
  - prism ..... 3-91
  - PRISM** command ..... 3-91
  - program sizes ..... 1-3
  - PROMPT**
    - see* **\$ PROMPT**
  - prompting
    - for parameter values ..... 2-14
    - in command input files ..... 2-23
  - properties ..... 3-70
  - pyramid ..... 3-91
- Q-
- Q** ..... 5-113, 5-114, 5-115
  - quadratic elements ..... 3-106, 4-81
  - quadratic extrusions ..... 4-51, 4-71
  - QUIT** sub-command
    - CONDUCTOR** command ..... 4-48
    - IDEAS** command ..... 4-123
    - SOLVERS** command ..... 4-167
- R-
- R, Z** ..... 5-113
  - racetrack ..... 3-44, 3-93, 4-29
  - RACETRACK** command ..... 3-92
  - radiation ..... 3-29
  - radiation boundary condition
    - see* boundary conditions
    - radiation
  - radio-buttons ..... 4-7
  - READ**
    - see* **\$ READ**

**READ** command ..... 4-149  
**READ** sub-command  
     **IDEAS** command ..... 4-124  
 redo ..... 3-121  
**REDO** command ..... 3-94  
 reduced scalar potential ..... 4-79, 4-80  
     interface ..... 4-80, 4-81  
 reflect ..... 3-119  
 regularise ..... 3-49  
 relaxation factor ..... 3-19  
 removing objects ..... 3-49  
**RENAME** command ..... 3-95  
 RHS integrals ..... 3-20, 4-80, 4-169  
 rotate ..... 3-119  
 rotational motion ..... 4-155  
 rotational velocity ..... 3-127  
 rubber-box ..... 4-4  
 Runge-Kutta ..... 3-21, 4-169  
     updates ..... 3-20

### -S-

**SAVE** command ..... 3-96  
 save picture ..... 3-88, 5-72  
 saving data ..... 3-96  
 SCALA ..... 3-20, 4-81, 4-88, 4-155, 5-12  
 scalar potential boundary condition  
     *see* boundary conditions  
     potential  
 scalar property ..... 4-81, 4-82  
 scale ..... 3-119  
 scroll bars ..... 4-5, 4-8  
**SELECT** command  
     modeller ..... 3-98  
     post processor ..... 5-80  
 selecting ..... 4-3  
 selecting parts for display ..... 3-99  
**SET** command ..... 5-85  
**SETTINGS** sub-command of the **SOLV-ERS** command ..... 4-167  
**SHOW** command ..... 5-88  
 simple updates ..... 3-20

simulation title ..... 3-42  
 simulations ..... 4-156  
 single filament conductors ..... 4-25  
 single valued potential ..... 4-79, 4-80  
 slider bars ..... 4-8  
**SLIP** command ..... 4-152  
 slip surface ..... 4-89, 4-152  
 solenoid ..... 3-44, 3-104  
**SOLENOID** command ..... 3-103  
 solenoids ..... 4-27  
**SOLVERS** command  
     modeller ..... 3-105  
     pre processor ..... 4-153  
 SOPRANO ..... 1-2, 3-20, 4-88, 4-155, 5-12  
 source current density ..... 5-12  
 source magnetic field strength ..... 5-12  
 space charge beam ..... 4-155  
 sphere ..... 3-107  
**SPHERE** command ..... 3-107  
 steady-state ac ..... 4-25, 4-155, 4-159  
 stored energy  
     integral ..... 5-47  
 straight ..... 3-44, 3-109  
 straight bars ..... 4-35  
**STRAIGHT** command ..... 3-108  
**STRING**  
     *see* \$ **STRING**  
 Subdivision Modification Mode ..... 4-141  
 subtracting bodies ..... 3-40  
 superscripts ..... 2-11  
**SURFACE** command ..... 5-90  
**SWEEP** command ..... 3-110  
 sweeping ..... 3-111  
 symmetry ..... 3-20, 3-82, 4-169  
     conductors ..... 4-26  
     recreating in post processor ..... 5-22  
 symmetry boundary condition ..... 3-29  
     *see* boundary conditions  
     symmetry  
 system variables ..... 5-10, 5-93

system variables for the **VIEW** command ..  
5-113

**SYSVARIABLE** command ..... 5-92

### -T-

**TABLE** command

post processor ..... 5-97

pre processor ..... 4-170

tangential constant perimeter end ..... 3-113

tangential electric ..... 3-29

tangential field boundary condition

*see* boundary conditions

tangential field

tangential magnetic ..... 3-29

tangential unit vector ..... 5-12

**TANGENTIALCPE** command ..... 3-112

text function ..... 2-11

text output modes ..... 2-22

**THREED** command

post processor ..... 3-114, 5-101

pre processor ..... 4-171

time ..... 5-14

time of flight ..... 5-113, 5-114

time stepping ..... 3-21, 4-169

timers ..... 4-8

time-step ..... 3-20

time-table files ..... 4-161

**TITLE** command

post processor ..... 3-116, 5-100

pre processor ..... 4-176

titles ..... 3-116, 4-159, 4-176, 5-100

**TODAY** ..... 2-21

**TOF** ..... 5-113, 5-114

tolerance ..... 3-18

conductor fields ..... 4-25

group operations ..... 4-66

topology problems ..... 3-37

torque ..... 5-15, 5-16, 5-28, 5-62

torus ..... 3-117

**TORUS** command ..... 3-117

**TOSCA** ..... 1-2, 3-20, 4-88, 4-155, 5-12

total scalar potential ..... 4-79, 4-80, 4-81

**TRACK** command ..... 5-104

**TRANSFORM** command

modeller ..... 3-118

pre processor ..... 4-178

transformations ..... 3-82, 3-119

transforming base plane coordinates .... 4-65

transient ..... 4-25, 4-155, 4-160

translate ..... 3-119

**TXBEAM, TYBEAM, TZBEAM** ..... 5-114

### -U-

undo ..... 3-121

**UNDO** command ..... 3-121

union ..... 3-40

units ..... 3-106

analysis data ..... 4-157

BH data ..... 4-16

post processing ..... 5-12, 5-109

**UNITS** command ..... 5-107

upwinding ..... 3-20, 4-169

user interface ..... 2-1

user variables ..... 2-19, 3-122

### -V-

**VARIABLE** command ..... 3-122

variables ..... 3-122

character ..... 2-21

in expressions ..... 2-8, 2-9

in free format input ..... 2-16

numerical ..... 2-19

system ..... 5-10, 5-93

**VECTOR** command ..... 3-123

vector maps ..... 5-69, 5-119

vector potential ..... 4-80, 4-81

vector potential boundary condition

*see* boundary conditions

vector potential

vector potential interface ..... 4-81

vector property ..... 4-82, 4-86

velocity ..... 3-127, 4-82, 4-85, 5-12

- VELX ..... 5-113, 5-114
  - VELY ..... 5-113, 5-114
  - VELZ ..... 5-113, 5-114
  - VERTEXDATA command ..... 3-125
  - vertical menu ..... 4-3
  - VFDIR ..... 2-11, 2-29, 4-7, 4-16
  - VFGRAPHICS ..... 4-95
  - VFINV ..... 4-96
  - VFWINDOWH ..... 4-96
  - VFWINDOWW ..... 4-96
  - VIEW command ..... 5-110
  - voltage ..... 3-29
  - voltage boundary condition
    - see boundary conditions
    - voltage
  - volume ..... 5-15, 5-47
    - label ..... 3-127
    - properties ..... 3-35, 3-127
  - VOLUME command
    - modeller ..... 3-126
    - post processor ..... 5-116
  - VX, VY and VZ ..... 2-6, 5-10
- W-
- WCS ..... 3-128
  - WCS command ..... 3-128
  - WHILE
    - see \$ WHILE
  - WINDOW command
    - modeller ..... 3-129
    - post processor ..... 5-119
  - working coordinate system ..... 3-69, 3-128
  - WRITE
    - see \$ WRITE
  - WRITE command ..... 4-179
  - WRITE sub-command
    - CONDUCTOR command ..... 4-48
- X-
- X, Y ..... 5-113, 5-114
  - X0BEAM, Y0BEAM, Z0BEAM ..... 5-114
- xpm file format ..... 3-88, 5-72