

# FrontISTR Analysis Flow Manual

## FrontISTR Commons

November 5, 2020

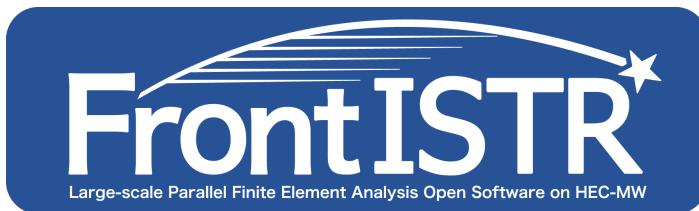
## Contents

<b>1</b>	<b>FrontISTR Analysis Flow Manual</b>	<b>2</b>
1.1	Manuals . . . . .	2
1.2	List of description on this manual . . . . .	2
1.3	Analysis Flow and Input/Output File . . . . .	3
1.3.1	Analysis Flow . . . . .	3
1.3.2	Overall Control Data . . . . .	4
1.3.3	Mesh Data . . . . .	4
1.3.4	Analysis Control Data . . . . .	4
1.3.5	Output File . . . . .	5
1.3.6	Execution Procedure . . . . .	5
1.3.7	Element Library . . . . .	6
1.3.8	Material Data . . . . .	11
1.3.9	Nonlinear Static Analysis . . . . .	15
1.4	Step control . . . . .	15
1.4.1	About time on analysis . . . . .	15
1.4.2	Control for static analysis . . . . .	16
1.4.3	Control for dynamic analysis . . . . .	18
1.5	Overall Control Data . . . . .	19
1.5.1	Outline of Overall Control Data . . . . .	19
1.5.2	Input Rules . . . . .	19
1.5.3	Header List . . . . .	19
1.6	Single Domain Mesh Data . . . . .	23
1.6.1	Outline of Single Mesh Data . . . . .	23
1.6.2	Input Rules . . . . .	23
1.6.3	Header List of Single Domain Mesh Data . . . . .	23
1.7	Analysis Control Data . . . . .	42
1.7.1	Outline of Analysis Control Data . . . . .	42
1.7.2	Input Rules . . . . .	43
1.7.3	Analysis Control Data . . . . .	44
1.7.4	Solver Control Data . . . . .	55
1.7.5	Post Process (Visualization) Control Data . . . . .	56
1.7.6	Details of Analysis Control Data Parameters . . . . .	64
1.7.7	Input of User Defined Material . . . . .	104
1.7.8	Subroutine regarding Elastoplasticity Deformation ( <i>uyield.f90</i> ) . . . . .	104
1.7.9	Subroutine regarding Elastic Deformation ( <i>uelastic.f90</i> ) . . . . .	104
1.7.10	Subroutine regarding User Defined Materials ( <i>umat.f</i> ) . . . . .	105
1.7.11	Process Subroutine of User Defined External Load ( <i>uload.f</i> ) . . . . .	106
1.8	<b>hecmw_part1</b> . . . . .	106
1.8.1	Domain segmentation technique . . . . .	106
1.8.2	Domain division type . . . . .	107
1.8.3	Files required for execution . . . . .	113
1.8.4	execution method . . . . .	114
1.8.5	Application examples . . . . .	115
1.8.6	error message . . . . .	121
1.9	<b>hecmw_vis1</b> . . . . .	124

1.10 hec2rcap . . . . .	124
1.11 rconv . . . . .	124
1.12 rmerge . . . . .	124

## 1 FrontISTR Analysis Flow Manual

This software is the outcome of “Research and Development of Innovative Simulation Software” project supported by Research and Development for Next-generation Information Technology of Ministry of Education, Culture, Sports, Science and Technology. We assume that you agree with our license agreement of “MIT License” by using this software either for the purpose of profit-making business or for free of charge. This software is protected by the copyright law and the other related laws, regarding unspecified issues in our license agreement and contact, or condition without either license agreement or contact.



Item	Content
Name of Software	FrontISTR
Version	5.1
License	MIT License
Corresponding Clerks	FrontISTR Commons2-11-16 Yayoi, Bunkyo-ku, Tokyo c/o Institute of Engineering Innovation, School of Engineering E-mail : support@frontistr.com

### 1.1 Manuals

- Introduction
- How to install
- Theory
- User's manual
- Tutorial
- FAQ

This manual describes the analysis method by the finite element method (FEM) used in FrontISTR.

Regarding the stress analysis method of solids, the infinitesimal deformation linear elasticity static analysis method is described first, and geometric nonlinear analysis method and elastoplasticity analysis method which are required when handling finite deformation problems are described next. Furthermore, a summarized evaluation method of the fracture mechanics parameters which can be acquired using the results of the stress analysis by FEM is described. Finally, the eigenvalue analysis and heat conduction analysis method is described.

### 1.2 List of description on this manual

- PDF
- Analysis Flow
- Input File
  - Overall Control Data
  - Mesh Partitioning Data
  - Mesh Data
  - Analysis Control Data
  - Visualization Control Data
- Output File

- Log Data
- Restart Data
- Visualization Data
- User Subroutines

## 1.3 Analysis Flow and Input/Output File

### 1.3.1 Analysis Flow

The input and output file of the structural analysis code FrontISTR is shown in Figure 3.1.1.

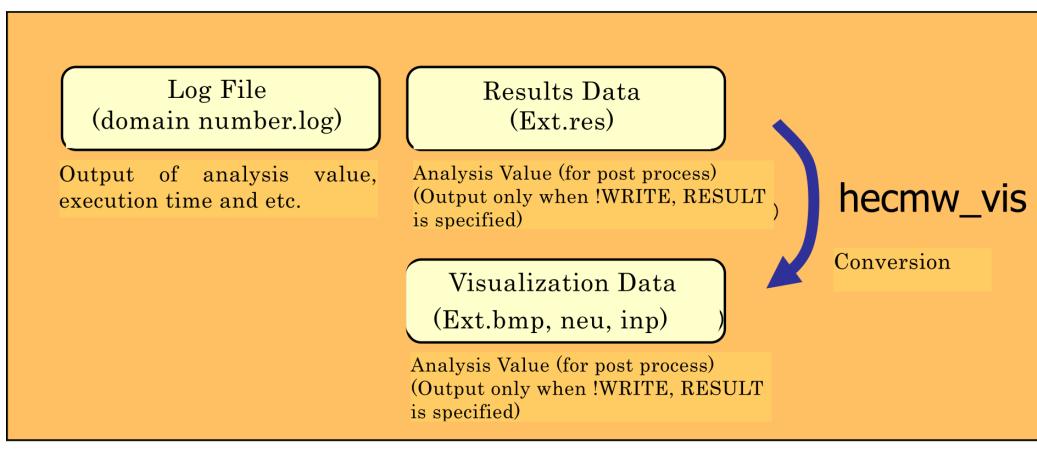
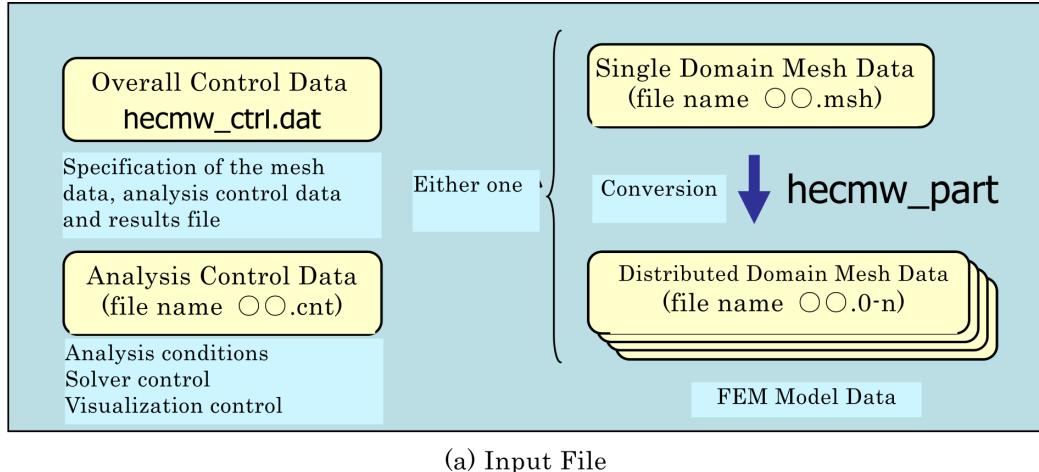


Figure 1: FrontISTR Input/Output File

### Figure 3.1.1:FrontISTR Input/Output File

FrontISTR requires three files, such as the overall control data, mesh data and analysis control data as input files.

When analyzing the overall model with a single CPU, the single domain mesh file is used. When performing parallel execution with multiple CPUs, the distributed domain mesh data as a result of performing domain partitioning of the single domain mesh data in advance by the hecmw\_part program which is the partitioner of *HEC-MW* in used.

The overall control data, analysis control data and single domain mesh data are text data. The user can create and edit using the proper editor, according to the description in this manual. The user can also create files using neu2fstr which converts a neutral file (\*.neu) supported by Femap which is a pre-post processor sold commercially as an attachment tool for FrontISTR, into FrontISTR input data.

Executing FrontISTR will output the log data file, results data file and visualization data. The existence and content of these outputs, depends on the description and analysis content in the analysis control file.

After FrontISTR is executed, the visualization data can also be created from the created results file by the hecmw\_vis program which is a tool included in *HEC-MW*.

### 1.3.2 Overall Control Data

This file specifies the input file and results output file of the mesh data and analysis control data.

The details of the overall control data are described in Chapter 5.

#### (Example)

```
!MESH, NAME=fstrMSH ,TYPE=HECMW-DIST
```

Definition of header of the distributed mesh data file (mandatory in the domain distribution model) : Foo\_P16

```
!MESH, NAME=fstrMSH ,TYPE=HECMW-ENTIRE
```

Definition of mesh data file name (mandatory in the single domain model) : Foo.msh

```
!CONTROL,NAME=fstrCNT
```

Definition of analysis control data file name (mandatory) : Foo.cnt

```
!RESULT,NAME=fstrRES ,IO=OUT
```

Definition of analysis results data file name (arbitrary) : Foo.res

```
!RESULT,NAME=vis_out ,IO=OUT
```

Definition of visualization file name (arbitrary) : Foo.vis

### 1.3.3 Mesh Data

This file defines the finite element mesh, and defines the material data and section data. This file also defines the group data used in analysis control data.

The details of the mesh data are described in Chapter 6

#### (Example)

```
!HEADER                                ——— Setting of mesh title
  TEST MODEL A361
!NODE                                 ——— Definition of node coordinates
  0.0 ,0.0 ,0.0
!ELEMENT,TYPE=361                      ——— Definition of element connectivity
  1001 ,1 ,2 ,3 ,4 ,5 ,6 ,7 ,8
!SECTION,TYPE=SOLID,EGRP=ALL,MATERIAL=M1 ——— Definition of section data
!MATERIAL,NAME=M1,ITEM=1                ——— Definition of material data
!ITEM=1,SUBITEM=2
  4000. ,      0.3
!NGROUP,NGRP=FIX ,GENERATE           ——— Definition of node group
  1001 ,1201 ,50
!EGROUP,EGRP=TOP,GENERATE            ——— Definition of element group
  1001 ,1201 ,1
!END
```

### 1.3.4 Analysis Control Data

This file defines analysis control data, such as the type of analysis, displacement boundary conditions, concentrated load and etc. Control of solver and the control data of the visualizer are also included in the analysis control data.

The details of the analysis control data described in Chapter 7

#### (EXAMPLE)

```

!! Analysis Type
!SOLUTION,TYPE=STATIC           ----- Specification of analysis type
!! Analysis control data
!BOUNDARY                         ----- Definition of displacement boundary conditions
  FIX,1,3,0.0
!CLOAD                            ----- Definition of concentrated load conditions
  CL1,1,-1.0
!DLOAD                            ----- Definition of distributed load conditions
  ALL,BX,1.0
!REFTEMP                           ----- Definition of reference temperature
  20.0
!TEMPERATURE                       ----- Definition of heat load (temperature) coditions
  ALL,100.0
!! Solver Control Data
!SOLVER,METHOD=CG,PRECOND=1,TIMELOG=YES,ITERLOG=YES ----- Control of Solver
  10000,2
  1.0e-8,1.0,0.0
!! Post Control Data
!WRITE,RESULT                      ----- Analysis results data output
!WRITE,VISUAL                      ----- Visualizer control
!! Visualizer
!visual                            ----- Hereinafter , the control data of the visualizer
!surface_num=1
!surface_style=1
!END

```

### 1.3.5 Output File

When the execution is completed, the log file (Ext.log) will be output. The analysis results file (Ext.res) for visualization will also be output by specifying the output.

The contents of the log files shown in the following will be output.

- Displacement, strain, Max/Min values of stress component
- Eigenvalues
- Engenvector values

### 1.3.6 Execution Procedure

#### 1.3.6.1 1. Preparation of FrontISTR

Save the main body of FrontISTR (Linux ver.: fistr1 , Windows ver. : fistr1 .exe) in the path directory, or the current direct directory at the time of execution.

#### 1.3.6.2 2. Preparation of Input Files

Prepare three types of input files hecmw\_ctrl.dat, analysis control and (single or distributed domain) mesh data, and enter the file name (pathname) of the analysis control data and mesh data in hecmw\_ctrl.dat. If necessary, also specify the analysis results data file and the visualization data file.

#### 1.3.6.3 3. Execution of Single Domain Analysis

Start the Linux terminal or the command prompt of Windows, move the current directory to the directory with the input file, and execute the analysis as follows (however, > refers to the prompt).

Example : In the case of Linux

> ./fistr1

Example : In the case of Windows

> fistr1

#### 1.3.6.4 4. Parallel Execution on Linux

In the Linux ver., the MPI must be installed in advance, and compiled for parallel execution. For details of the compiling method, refer to the Installation Manual. The execution depends on the settings of the execution environment of MPI. An example of execution int four domains is shows in the following.

```
> mpirun -np 4 ./fistr1
```

#### 1.3.6.5 5. Parallel Execution on Windows

In the Windows ver., it is necessary to download the library of MPICH2 from the following URL and install. For the method of parallel execution, refer to MPICH2 Manual.

- <http://www-unix.mcs.anl.gov/mpi/mpich/>

#### 1.3.6.6 6. Execution of Parallel Domain Contact Analysis (Note for users from Ver.3.x)

Until Ver.3.x, only when performing contact analysis, users had to always specify single domain mesh file as input mesh no matter if it is serial or parallel analysis. Starting from Ver.5.x, this exception was removed; i.e. users have to specify single domain mesh file as input mesh when performing serial contact analysis, distributed domain mesh file as input mesh when performing parallel contact analysis, just like other analysis types. ## Element Library and Material Data

#### 1.3.7 Element Library

In FrontISTR, the element groups shows in Table 4.1.1 can be used for analysis. Since HEC-MW is used to input the mesh data into FrontISTR, the following descriptions of the element library is based on the description of HEC-MW. The element library is shown in Figure 4.1.1, and element connectivity and the definitions of the surface number are shown in Figure 4.1.2.

Table 4.1.1: Element Library List

Element Types	Element No.	Description
Line element	111	Two node link element
	112	Three node link element
Plane element	231	Three node triangular element
	232	Six node triangular quadratic element
	241	Four node quadrilateral element
	242	Eight node quadrilateral quadratic elment
Solid element	301	Two node truss element
	341	Four node tetrahedral element
	342	Ten node tetrahedral quadratic element
	351	Six node pentahedral element
	352	Fifteen node pentahedral quadratic element
	361	Eight node hexahedral element
	362	Twenty node hexahedral quadratic element
Interface element	541	Quadrilateral cross section linear element
	542	Quadrilateral cross section quadratic element
Beam element	611	Two node beam element
	641	Two node beam element (with four 3-dof nodes)
Shell element	731	Three node three-dimensional linear element
	732	Six node three-dimensional quadratic element
	741	Four node three-dimensional linear element
	743	Nine node three-dimensional quadratic element
	761	Three node three-dimensional linear element (with six 3-dof nodes)
	781	Four node three-dimensional linear element (with eight 3-dof nodes)

Figure 4.1.1: Element Library

##### 1.3.7.1 (Line Element)

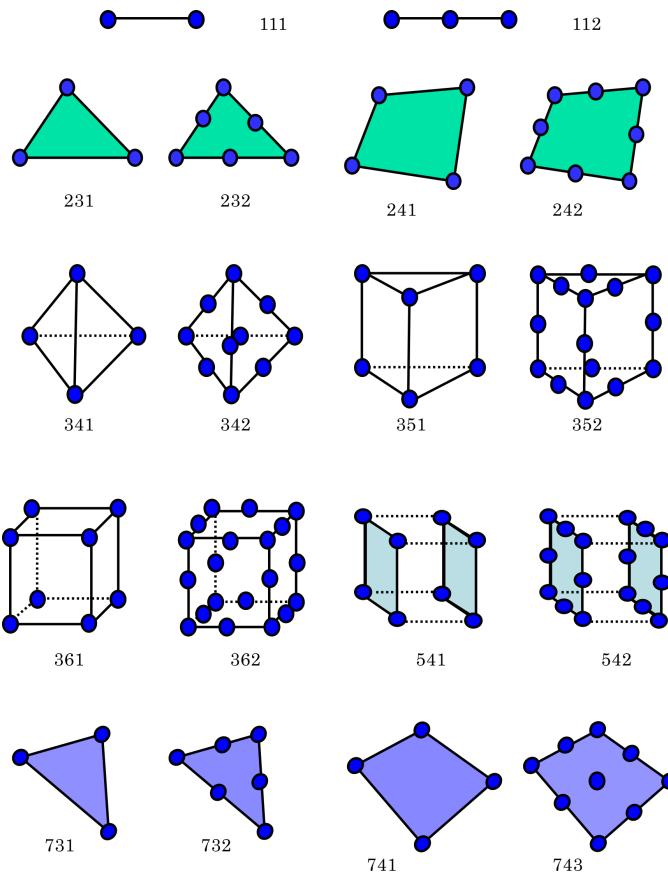


Figure 2: Element Library



Figure 3: Line Element

### 1.3.7.2 (Triangular Plane Element)

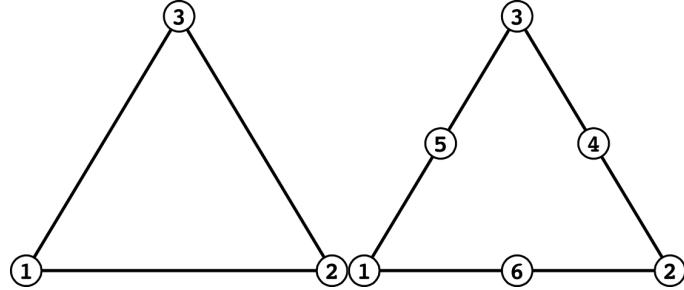


Figure 4: Triangular Plane Element

Surface No.	Linear	Quadratic
1	1 - 2	1 - 6 - 2
2	2 - 3	2 - 4 - 3
3	3 - 1	3 - 5 - 1

### 1.3.7.3 (Quadrilateral Plane Element)

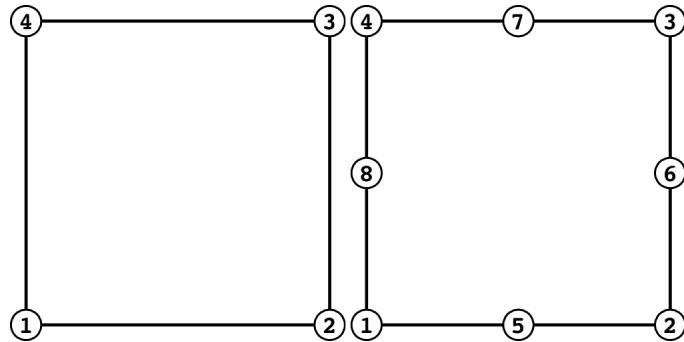


Figure 5: Quadrilateral Plane Element

Surface No.	Linear	Quadratic
1	1 - 2	1 - 5 - 2
2	2 - 3	2 - 6 - 3
3	3 - 4	3 - 7 - 4
4	4 - 1	4 - 8 - 1

### 1.3.7.4 (Tetrahedral Element)

Surface No.	Linear	Quadratic
1	1 - 2 - 3	1 - 7 - 2 - 5 - 3 - 6
2	1 - 2 - 4	1 - 7 - 2 - 9 - 4 - 8
3	2 - 3 - 4	2 - 5 - 3 - 10 - 4 - 9
4	3 - 1 - 4	3 - 6 - 1 - 10 - 4 - 8

### 1.3.7.5 (Pentahedral Element)

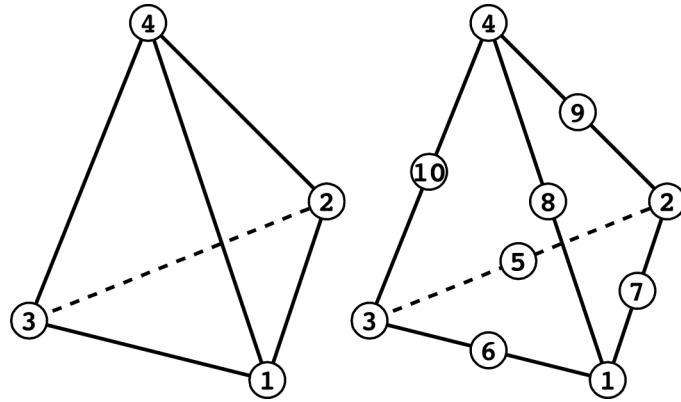


Figure 6: Tetrahedral Element

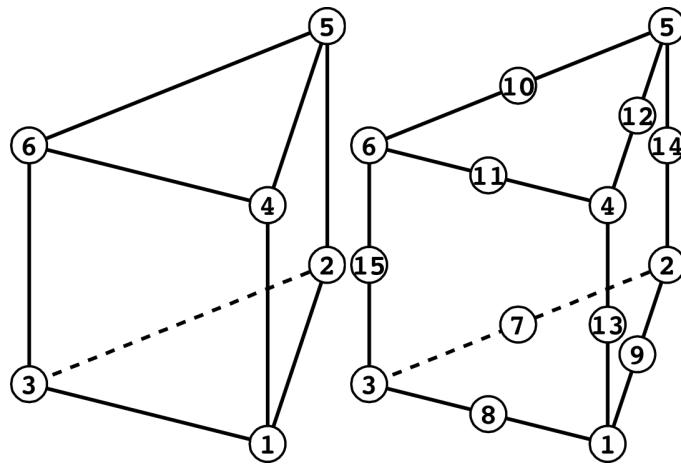


Figure 7: Pentahedral Element

Surface No.	Linear	Quadratic
1	1 - 2 - 3	1 - 9 - 2 - 7 - 3 - 8
2	4 - 5 - 6	4 - 12 - 5 - 10 - 6 - 11
3	1 - 2 - 5 - 4	1 - 9 - 2 - 14 - 5 - 12 - 4 - 13
4	2 - 3 - 6 - 5	2 - 7 - 3 - 15 - 6 - 10 - 5 - 14
5	3 - 1 - 4 - 6	3 - 8 - 1 - 13 - 4 - 11 - 6 - 15

#### 1.3.7.6 (Hexahedral Element)

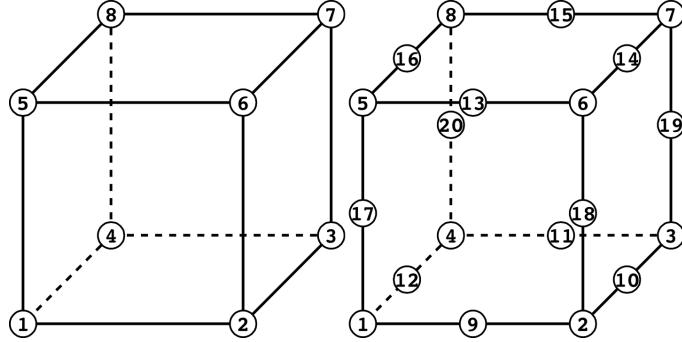


Figure 8: Hexahedral Element

Surface No.	Linear	Quadratic
1	1 - 2 - 3 - 4	1 - 9 - 2 - 10 - 3 - 11 - 4 - 12
2	5 - 6 - 7 - 8	5 - 13 - 6 - 14 - 7 - 15 - 8 - 16
3	1 - 2 - 6 - 5	1 - 9 - 2 - 18 - 6 - 13 - 5 - 17
4	2 - 3 - 7 - 6	2 - 10 - 3 - 19 - 7 - 14 - 6 - 18
5	3 - 4 - 8 - 7	3 - 11 - 4 - 20 - 8 - 15 - 7 - 19
6	4 - 1 - 5 - 8	4 - 12 - 1 - 17 - 5 - 16 - 8 - 20

#### 1.3.7.7 (Beam Element)



Figure 9: Beam Element

#### 1.3.7.8 (Beam Element with 3-dof nodes)



Figure 10: Beam Element with 3-dof nodes

Nodes 1 and 2 for translational DOF, nodes 3 and 4 for rotational DOF.

#### 1.3.7.9 (Triangular Shell Element)

Surface No.	Linear	Quadratic
1	1 - 2 - 3 [front]	1 - 6 - 2 - 4 - 3 - 5 [front]
2	3 - 2 - 1 [back]	3 - 4 - 2 - 6 - 1 - 5 [back]

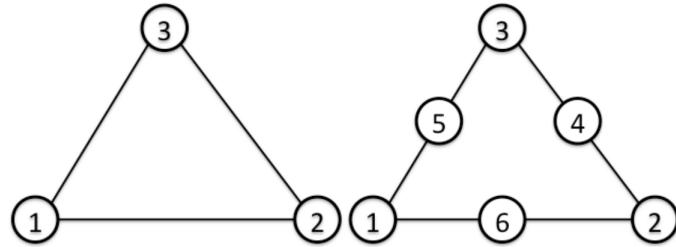


Figure 11: Triangular Shell Element

#### 1.3.7.10 (Triangular Shell Element with 3-dof nodes)

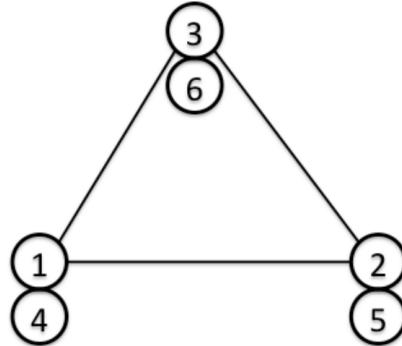


Figure 12: Triangular Shell Element with 3-dof nodes

Node 1, 2 and 3 for translational DOF, nodes 4, 5 and 6 for rotational DOF.

Surface No.	Linear
1	1 - 2 - 3 [front]
2	3 - 2 - 1 [back]

#### 1.3.7.11 (Quadrilateral Shell Element)

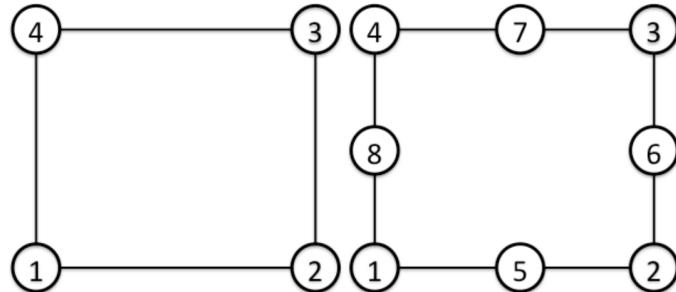


Figure 13: Quadrilateral Shell Element

Surface No.	Linear	Quadratic
1	1 - 2 - 3 - 4 [front]	1 - 5 - 2 - 6 - 3 - 7 - 4 - 8 [front]
2	4 - 3 - 2 - 1 [back]	4 - 7 - 3 - 6 - 2 - 5 - 1 - 8 [back]

#### 1.3.7.12 (Quadrilateral shell Element with 3-dof nodes)

Nodes 1, 2, 3 and 4 for translational DOF, nodes 5, 6, 7 and 8 for rotational DOF.

Surface No.	Linear
1	1 2 - 3 - 4 [front]
2	4 - 3 - 2 - 1 [back]

Figure 4.1.2: Connectivity and Surface Number

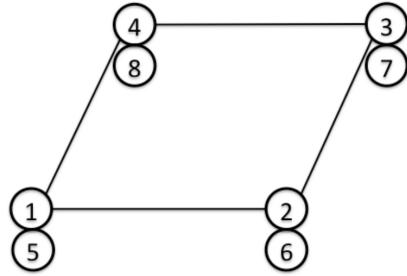


Figure 14: Quadrilateral shell Element with 3-dof nodes

and define the Young's modulus, Poisson's ratio density and the coefficient of linear expansion for each element. The property value of these materials is defined in the header !SECTION and !MATERIAL of the data. An example of the definition is in the following.

**(Example)**

```
!! Definition of SECTION
!SECTION,TYPE=SOLID,EGRP=ALL,MATERIAL=M1
```

In the above !SECTION, the material data name of the element belonging to the group name=ALL in the solid type element is defined as M1.

Next, the definition method of the material data is shown in the following.

**(Example)**

```
!! Intention of defining three types of property values in the material name M1
!MATERIAL,NAME=M1,ITEM=3
```

```
!! Young's modulus and Poisson's ratio are defined in !ITEM=1 (mandatory)
!ITEM=1,SUBITEM=2
4000.0,0.3
```

```
!! The mass density must be defined in !ITEM=2 (mandatory in the case of ITEM=3)
!ITEM=2
8.0102E-10
```

```
!! The coefficient of linear expansion must be defined in !ITEM=3
!ITEM=3
1.0E-5
```

As long as each item number and the type of physical property are compatible, and the ITEM number is correct, it can be defined in any order. However, ITEM=1 must be defined in the order of the Young's modulus and Poisson's ratio.

### 1.3.8.2 Heat Conduction Analysis

In the heat conduction analysis of FrontISTR, isotropic material taking into consideration the temperature dependency can be used. Regarding the property values to be defined, the density, specific heat and thermal conductivity are defined for the link, plane surface, solid and shell element, and the gap heat transfer coefficient and the gap radiation factor are defined in the interface element. An example of the definition method of these property values is shown in the following.

#### 1.3.8.2.1 (1) In the case of link, plane surface and solid element

Defined in the !SECTION and !MATERIAL header.

**(Example)**

```
!! Definition of section
!SECTION,TYPE=SOLID,EGRP=ALL,MATERIAL=M1
```

In the above !SECTION, the material data name of the element belonging to the group name=ALL in the solid type element is defined as M1. The following shows the definition method of the material data.

**(Example)**

```
!! Intension of defining three types of property values
!! in the material of material name M1
!MATERIAL,NAME=M1,ITEM=3

!! The density and temperature are defined in !ITEM=1 (mandatory)
!ITEM=1,SUBITEM=1
 7850., 300.
 7790., 500.
 7700., 800.

!! The specific heat and temperature are defined in !ITEM=2 (mandatory)
!ITEM=2
 0.465, 300.
 0.528, 500.
 0.622, 800.

!! The thermal conductivity and temperature are defined in !ITEM=3 (mandatory)
!ITEM=3
 43., 300.
 38.6, 500.
 27.7, 800.
```

Since the type of physical properties are identified by each item number in FrontISTR, it can be defined in any order as long as both are consistent.

### 1.3.8.2.2 (2) In the case of an interface element

Defined in the !SECTION header. (Material data is not required)

**(Example)**

```
!! Definition of section
!SECTION,TYPE=INTERFACE,EGRP=GAP
 1.0, 20.15, 8.99835E-9, 8.99835E-9
```

In the above !SECTION, the gap parameter of the element belonging to the group name=GAP in the interface element is defined.

- 1st parameter : Gap width
- 2nd parameter : Gap heat transfer coefficient
- 3rd parameter : Gap radiation factor 1
- 4th parameter : Gap radiation factor 2

### 1.3.8.2.3 (3) In the case of a shell element

Defined in the !SECTION and MATERIAL header.

**(Example)**

```
!! Definition of section
!SECTION, TYPE=SHELL, EGRP=SH, MATERIAL=M2
 10.0, 5
```

In the above !SECTION, the shell characteristics of the element belonging to the group name=SH in the shell type element is defined.

- 1st parameter : Shell thickness
- 2nd parameter : Integration points in the thickness direction

This also means that the material data name of the material physical properties of the element which belong to this group is defined as M2.

For single layered isotropic shell, the definition method of the material physical properties is the same as in the case of the solid element. Refer to the description of the solid element.

In addition, another definition method stated below is available for multi-layered and/or orthotropic shell.

#### (Example) Isotropic, single-layered shell

```
!MATERIAL, NAME=M1, ITEM=1  
!ITEM=1,SUBITEM=4  
0, 200000, 0.3, 2.0
```

- 1st parameter : 0=isotropic
- 2nd parameter : Young's modulus
- 3rd parameter : Poisson's ratio
- 4th parameter : Shell thickness

#### (Example) Isotropic, double-layered shell

```
!MATERIAL, NAME=M1, ITEM=1  
!ITEM=1,SUBITEM=7  
0, 200000, 0.3, 2.0, 200000, 0.3, 2.0
```

- 1st parameter : 0=isotropic
- 2nd parameter : Young's modulus (1st layer)
- 3rd parameter : Poisson's ratio (1st layer)
- 4th parameter : Shell thickness (1st layer)
- 5th parameter : Young's modulus (2nd layer)
- 6th parameter : Poisson's ratio (2nd layer)
- 7th parameter : Shell thickness (2nd layer)

By specifying as many material constants as needed for the number of layers, the material is automatically treated as multi-layered shell. The total thickness of the shell is the sum of the thickness of all the layers. The layers are ordered from the top surface of the shell with respect to the normal direction.

#### (Example) Orthotropic, single-layered shell

```
!MATERIAL, NAME=M1, ITEM=1  
!ITEM=1, SUBITEM=9  
1, 28600., 0.15, 32.3, 28600., 12434., 12434., 12434., 0.0
```

- 1st parameter : 1=orthotropic
- 2nd parameter : Young's modulus E1
- 3rd parameter : Poisson's ratio n12
- 4th parameter : Shell thickness
- 5th parameter : Young's modulus E2
- 6th parameter : Shear modulus G12
- 7th parameter : Shear modulus G23
- 8th parameter : Shear modulus G13
- 9th parameter : Orthotropy angle

Orthotropy angle is specified in [degree].

#### (Example) Orthotropic, double-layered shell

```
!MATERIAL, NAME=M1, ITEM=1  
!ITEM=1, SUBITEM=17  
1, 28600., 28600., 0.15, 32.3, 12434., 12434., 28600., 12434., 12434., 12434., 0.0
```

- 1st parameter : 1=orthotropic
- 2nd parameter : Young's modulus E1 (1st layer)
- 3rd parameter : Poisson's ratio n12 (1st layer)
- 4th parameter : Shell thickness (1st layer)

- 5th parameter : : Young's modulus E2 (1st layer)
- 6th parameter : Shear modulus G12 (1st layer)
- 7th parameter : Shear modulus G23 (1st layer)
- 8th parameter : Shear modulus G13 (1st layer)
- 9th parameter : Orthotropy angle q (1st layer)
- 10th parameter : Young's modulus E1 (2nd layer)
- 11th parameter : Poisson's ratio n12 (2nd layer)
- 12th parameter : Shell thickness (2nd layer)
- 13th parameter : Young's modulus E2 (2nd layer)
- 14th parameter : Shear modulus G12 (2nd layer)
- 15th parameter : Shear modulus G23 (2nd layer)
- 16th parameter : Shear modulus G13 (2nd layer)
- 17th parameter : Orthotropy angle (2nd layer)

### 1.3.9 Nonlinear Static Analysis

In the nonlinear static analysis of FrontISTR, in addition to the method to define the !SECTION and !MATERIAL shown in item 4.2.1, !ELASTIC, !HYPERELASTIC, !PLASTIC and etc. in the analysis control data can also be defined. An example of the definition is shown in the following.

#### (Example) Definition of hyperelastic material

!MATERIAL

```
!! Definition of the Neo Hooke hyperelastic material
!HYPERELASTIC, TYPE=NEOHOOKE
```

```
!! C_10 and D are defined (mandatory)
1000.0, 0.00005
```

#### (Example) Definition of the elastoplastic material

!MATERIAL

```
!! Definition of the isotropic elastic material
!ELASTIC, TYPE=ISOTROPIC
```

```
!! The Young's modulus and the Poisson's ratio are defined (mandatory)
21000.0, 0.3
```

```
!! Definition of the Drucker-Prager plastic material
!PLASTIC, TYPE=DRUCKER-PRAGER
```

```
!! Viscosity, angle of friction and hardening factor are defined (mandatory)
500.0, 4.0, 10.0
```

## 1.4 Step control

### 1.4.1 About time on analysis

In the analysis of FrontISTR, following definitions of “time” are used:

- Current time : Total time from start of analysis.
- Step time : Time from start in each of step.
- Time span : Duration of time for each step.
- Relative time : Ratio of “Step time” to “Time span”.
- Time increment : Increment from “Current time” to “time at which the equilibrium is attained”.

Figure 4.7.1 Definition of terms for analysis time

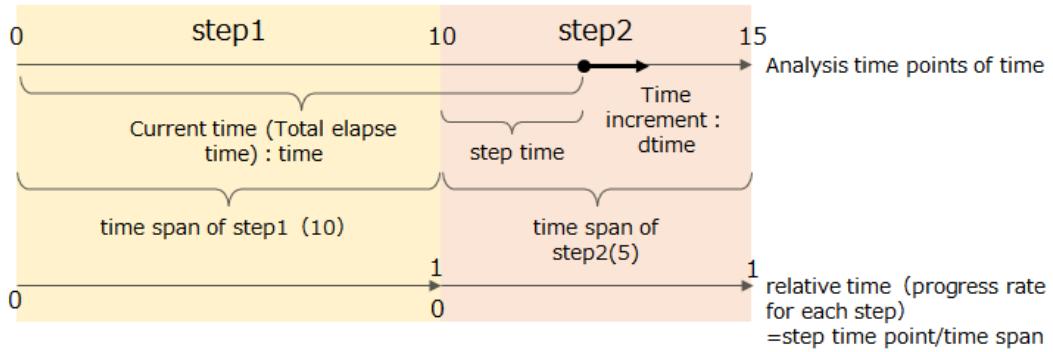


Figure 15: Definition of terms for analysis time

#### 1.4.2 Control for static analysis

A static analysis in FrontISTR consists of one or more (continuous) analysis steps. A set of boundary conditions is given in each step and the analysis step is solved with incremental solving strategy which is mentioned previously.

Below, iteration of analysis is called “step loop”, and incremental method is called “sub-step loop”.

Incremental control of static analysis could be chosen from the following solution.

- Analyze with fixed time increments. If convergence fails, the calculation will be stopped immediately.
- Automatic time increment and cutback. Depending on the convergence, change the increment width. If the calculation fails, recalculate after decreasing the increment width.

##### 1.4.2.1 Automatic time increment and cut-back

Flow is automatic time increment and cut-back is as follows(Figure 4.7.2).

Figure 4.7.2 Flow of automatic time increment and cut-back

Overview of program flows is as follows,

1. Loop from step 1 to final step(N\_step):
2. The time increment base `dtime_base` is determined from the current `dtime_base` and the convergence status in the previous sub-step. First time uses `initdt`.
3. The actual `dtime` is determined by "step end or the remaining time until the nearest specified output time and smaller of `dtime_base`".
4. Try calculation of equilibrium of forces at time+`dtime`.
5. If converge, the time is forward by `dtime`, when it fails, restore time and return to 2.
6. Step ends when time reaches step end time.

If any of the following cases occurs during the analysis, the nonlinear static analysis is considered to have failed and the program terminates with error:

- The number of sub-steps reaches the upper limit before time reaches the step end time.
- The time increment `dtime_base` falls below the time increment lower limit `mindt`.
- Analysis fails to converge consecutively for specified number (`N_C`) of times.

##### 1.4.2.2 Adjust time increment baseline `dtime_base`

The first `dtime_base` of the step is set to the value of the specified initial time increment `initdt`. Otherwise, it is set as follows according to the convergence status of the previous sub-step.

1. If analysis failed to converge and a cutback is made, the value `dtime_base` multiplied by `R_C` (reduction ratio of cutback).
2. When be a successful to converge
  1. Corresponds to the decrease condition : the value of `dtime_base` multiplied by `R_S` (decrease ratio).
  2. In the case of an increase condition, not a decrease condition : the smaller value of ‘`dtime_base`’ multiplied by ‘`R_L`’ (increase ratio) and the time increment upper limit `maxdt`.
  3. If neither : `dtime_base` dose not change.

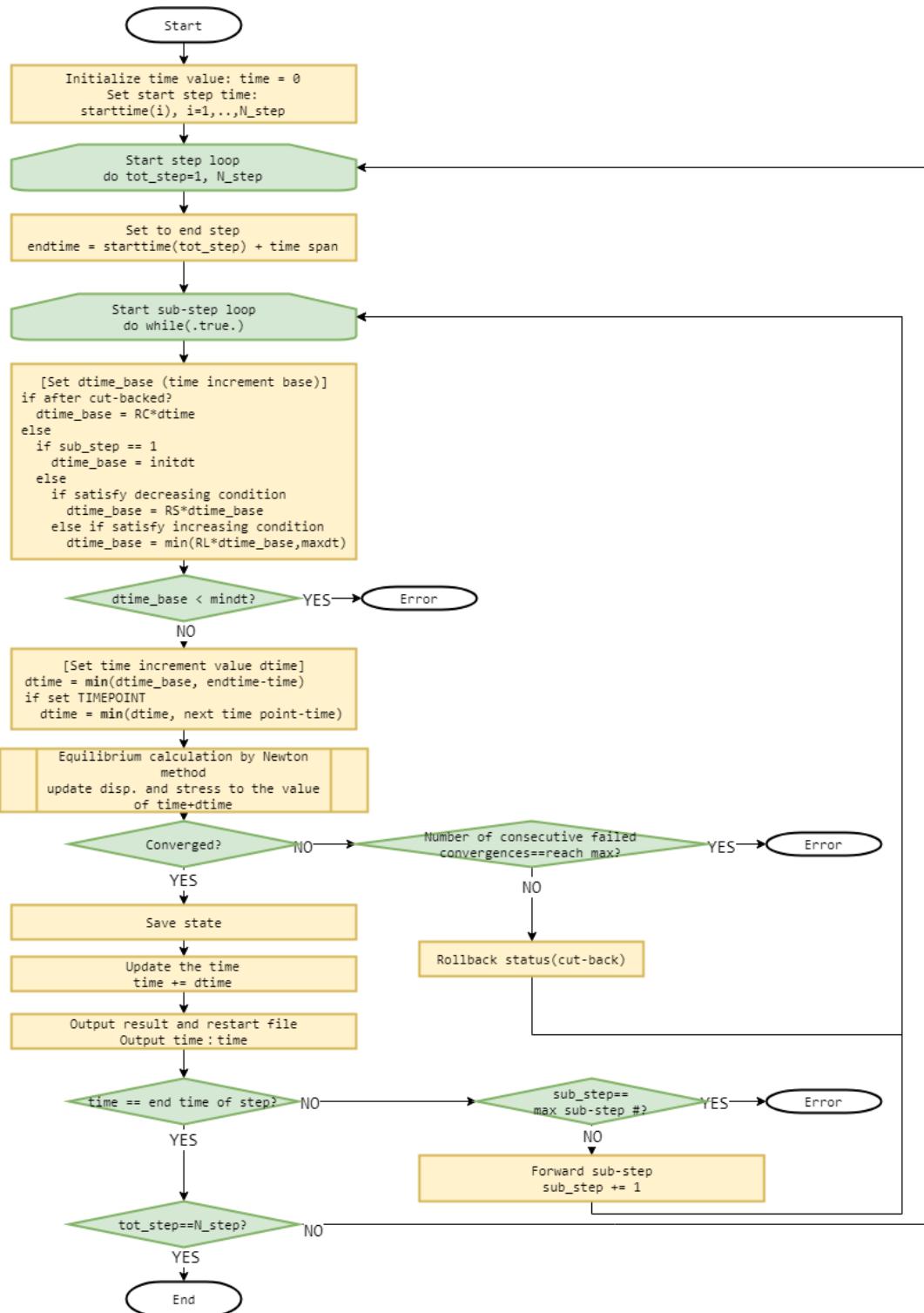


Figure 16: Flow of automatic time increment and cut-back

### 1.4.2.3 Increase / decrease conditions

When the automatic time incremental adjustment function determines to increase/decrease time increment, the following variables are used:

- N\_max : Maximum number of Newton iterations of previous substep.
- N\_sum : Total number of Newton iterations in the previous substep(Same as N\_max, when there is no repeated contact).
- N\_cont : Number of contact iterations in previous substep.

Time increment is decreased when both of the following are met:

- One of N\_max, N\_sum or N\_cont exceeds the respective threshold NS\_max, NS\_sum or NS\_cont.
- When the above conditions are satisfied by a substep that continues N\_S times or more.

Time increment is increased when both of the following are met:

- N\_max, N\_sum and N\_cont are all within their respective threshold NL\_max, NL\_sum and NL\_cont.
- The above condition is met by substep that continues more than N\_L times.

### 1.4.2.4 Specify calculation time and output time

With automatic time increments, the increments change with the convergence state, so it is not possible to determine in advance when to run the equilibrium calculation and result output. If this is inconvenient, you can specify a list of output times to perform the equilibrium calculation and result output at the desired time.

For steps that include a list of output times, dtime is adjusted so that the calculation is always performed at specified time.

### 1.4.2.5 Usage of function of automatic time increments and cut-back

Settings related to this function are made in the analysis control file.

The function of automatic time increment and cut-back are enabled specified by TYPE\_INC=AUTO on the !STEP card. For parameters that adjust the time increment, define !AUTOINC\_PARAM and specify it in each step using the !STEP, AUTOINCPARAM'. If not specified, the default value of!AUTOINC\_PARAM' is used.

The output time is specified for each step with the !STEP, TIMEPOINTS parameter after defining the time list with !TIME\_POINTS.

#### 1.4.2.5.1 Example

Enable automatic incremental adjustment and set initial time increment 0.01, step duration 2.5, time increment lower limit 1E-5, time increment upper limit 0.3, maximum substep number to 200. Specify AP1 for the auto-increment parameter set and 1.5, 2.7 and 3.9 for the output time.

```
!AUTOINC_PARAM, NAME=AP1
0.25, 10, 50, 10, 1
1.25, 1, 1, 1, 2
0.25, 5
!TIME_POINTS, TIME=TOTAL, TIME=,NAME=TP1
1.5
2.7
3.9
!STEP, INC_TYPE=AUTO, SUBSTEPS=200, AUTOINCPARAM=AP1, TIMEPOINTS=TP1
0.01, 2.5, 1E-5, 0.3
```

### 1.4.3 Control for dynamic analysis

In this development code, dynamic analysis direct time integration method consists of one analysis step.

Incremental control for dynamic analysis is only fixed increments and automatic time increments are not available.

## 1.5 Overall Control Data

### 1.5.1 Outline of Overall Control Data

The overall control data is for defining the file name of the input/output files for FrontISTR. The features of the overall control data file are as follows.

- This is an ASCII format file based on a free format.
- This file consists of a header which starts with “!” and the data following this.
- The order of description of the header is basically free.
- A “,” is used as a punctuation mark of the data.

### 1.5.2 Input Rules

The overall control data file consists of a header line, data line and a comment line. One header is always included in the header line.

**Header** The meaning of the data and data block is specified in the overall control data file.

When the head of the term starts with a “!”, it is considered to be a header.

**Header Line** The header and the parameter accompanying this are described in this line.

The header line must start with a header.

When a parameter is required, a “,” must be used to continue after that.

When the parameter takes on a value, use an “=” after the parameter and describe the value after that.

**Data Line** The data line starts from the next line of the header line, and the necessary data is described.

The data lines may be in multiple lines; however, this is determined according to the rules of the data description defined by each header.

There are cases where data lines are not required.

**Punctuation** A comma “,” is used as a punctuation of the data.

**Handling of Blanks** Blanks are disregarded.

**Name** Regarding the characters which can be used for the name, there is the underscore “\_”, hyphen “-”, and alphanumeric characters “a – z, A – Z, 0 – 9”; however, the first letter of the name must start with “\_”, or an alphabetic character “a – z, A – Z”.

There is no distinction between uppercase and lowercase letters, and all letters are internally handled as uppercase letters.

The maximum length of the name is 63 characters.

**File Name** Regarding the characters which can be used for the file name, there are the underscore “\_”, hyphen “-”, period “.”, slash “/”, and the alphanumeric characters “a – z, A – Z, 0 – 9”.

As long as there is no specific description, a path can be included in the file name. Both the relative path and the absolute path can be specified.

The maximum length of the file name is 1,023 characters.

**Floating Point Data** Exponents are optional. An “E” or “e” character must be added before the exponent.

The selection of “E” or “e” is optional. “D” or “d” can not be used.

**!!, # Comment Line** Lines starting with “!!” or “#” are considered to be comment lines, and are disregarded. A comment line can be inserted in any position in the file, and there are no restrictions on the number of lines.

### 1.5.3 Header List

The overall control data consists of the following headers.

Header Name	Contents
!CONTROL	Analysis control data definition
!MESH	Mesh data definition
!RESTART	Restart data definition
!RESULT	Analysis results data definition
!SUBDIR	Dividing to sub directories

In each header, there are data items which are compatible to the parameter and each header. Each of the above headers is described in the following with examples of data creation.

#### 1.5.3.1 (1) !CONTROL

Specifies the analysis control data file.

```
!CONTROL,NAME=<name>
file
```

##### Example of Use

```
!CONTROL, NAME=fstrCNT
myctrl.dat
```

Parameter	
NAME	Identifier (mandatory)

Parameter Name	Parameter Value	Contents
NAME	fstrCNT	Analysis control data

Parameter Name	Contents
file	Analysis control data file name (both the relative path and the absolute path can be specified. When the relative path is specified, it becomes the path from the current directory.)

### 1.5.3.2 (2) !MESH

Specifies the mesh data file.

```
!MESH, NAME=<name>, TYPE=<type> [ , optional parameter ]
fileheader
```

#Example of Use

```
!MESH, NAME=fstrMSH, TYPE=HECMW-DIST, REFINE=1
Mesh.in
```

Parameter	
NAME	Identifier (mandatory)
TYPE	Mesh type (mandatory)
IO	Input/output specification (omissible)
REFINE	Mesh subdivision specification (arbitrary)

Parameter Name	Parameter Value	Contents
NAME	fstrMSH	Solver input data
	part_in	Partitioner input data
	part_out	Partitioner output data
TYPE	HECMW-DIST	HEC-MW distribution mesh data
	HECMW-ENTIRE	HEC-MW single domain mesh data
IO	IN	For input (default)
	OUT	For output
REFINE	<integer>	Number of mesh subdivisions

Parameter Name	Contents
fileheader	Header of the mesh data file name (both the relative path and the absolute path can be specified. When the relative path is specified, it becomes the path from the current directory.)

Note:

The existence of IO parameters, or parameter values will have no affect on others. When the TYPE is HECMW-DIST, the end of the file name “.<rank>” is excluded for the file header specified in the data line.

#### 1.5.3.3 (3) !RESTART

Specifies the restart data file.

**!RESTART, NAME=<name>, IO=<io>**

fileheader

```
# Example of Use
!RESTART, NAME=restart-in, IO=IN
restart.in
```

Parameter Name	Contents
NAME	Identifier (mandatory)
IO	Input/output specification (mandatory)

Parameter Name	Parameter Value	Contents
NAME	<name>	Identifier
IO	IN	For input
	OUT	For output
	INOUT	Common to input/output

Parameter Name	Contents
fileheader	Header of the restart data file name (both the relative path and the absolute path can be specified. When the relative path is specified, it becomes the path from the current directory.)

Note:

The file name created by this definition is the file header+.<rank>.

#### 1.5.3.4 (4) !RESULT

Specifies the analysis results data file.

**!RESULT, NAME=<name> [ , optional parameter ]**

fileheader

```
# Example of Use
!RESULT, NAME=fstrRES, IO=OUT, TYPE=BINARY
result.out
```

Parameter	
NAME	Identifier (mandatory)
IO	Input/output specification (mandatory)
TYPE	Output format (omissible)

Parameter Name	Parameter Value	Contents
NAME	fstrRES	Solver output data, Visualizer input data
	fstrTEMP	Temperature input data (Result of heat analysis)
	vis_out	Visualizer output data
IO	IN	For input
	OUT	For output
TYPE	TEXT	Text format (default)
	BINARY	Binary format

Parameter Name	Contents
fileheader	Header of the analysis results data file name (both the relative path and the absolute path can be specified. When the relative path is specified, it becomes the path from the current directory.)

Note:

The file name created by this definition is the file header+.<rank>.

### 1.5.3.5 (5) !SUBDIR

Specifies storing input/output files into subdirectories

**!SUBDIR, ON [ , optional parameter ]**

# Example of Use

**!SUBDIR, ON, LIMIT=8000**

Parameter	
ON	Enable (mandatory)
LIMIT	Number of files (omissible)

Parameter Name	Parameter Value	Contents
ON	N/A	
LIMIT	<integer>	Maximum number of files per directory (default:5000)

Note:

The input/output data which are comprised of plural files are automatically stored into each subdirectory by this definition. Furthermore, if the number of ranks exceeds “LIMIT”, those files are divided to subdirectoris such as TRUNK0, TRUNK1.

## 1.6 Single Domain Mesh Data

### 1.6.1 Outline of Single Mesh Data

In order to acquire the mesh data of FrontISTR, there are two methods, such as the method to input the single domain mesh data file, and the method to input the distributed mesh data file to perform parallel processing. Single domain mesh data is described in this section.

The features of single domain mesh data are as follows.

- This is an ASCII format file based on a free format.
- This file consists of a header which starts with “!” and the data following this.
- The order of description of the header is basically free.
- A “,” is used as a punctuation mark of the data.

### 1.6.2 Input Rules

The single domain mesh data file consists of a header line, data line and a comment line. One header is always included in the header line.

**Header** The meaning of the data and data block is specified in the single domain mesh data file.

When the head of the term starts with a “!”, it is considered to be a header.

**Header Line** The contents of the header and the parameter accompanying this are described in this line.

The header line must start with a header. When a parameter is required, a “,” must be used to continue after that.

When the parameter takes on a value, use an “=” after the parameter and describe the value after that. The header line can not be described in more than two lines.

**Data Line** The data line starts after the header line, and the necessary data is described.

The data lines may be in multiple lines; however, this is determined according to the rules of the data description defined by each header.

There are cases where data lines are not required.

**Punctuation** A comma “,” is used as a punctuation of the data.

**Handling of Blanks** Blanks are disregarded.

**Name** Regarding the characters which can be used for the name, there is the underscore “\_”, hyphen “-”, and alphanumeric characters “a – z, A – Z, 0 – 9”; however, the first letter of the name must start with “\_”, or an alphabetic character “a – z, A – Z”. There is no distinction between uppercase and lowercase letters, and all letters are internally handled as uppercase letters.

The maximum length of the name is 63 characters.

**File Name** Regarding the characters which can be used for the file name, there are the underscore “\_”, hyphen “-”, period “.”, slash “/”, and the alphanumeric characters “a – z, A – Z, 0 – 9”.

As long as there is no specific description, a path can be included in the file name. Both the relative path and the absolute path can be specified.

The maximum length of the file name is 1,023 characters.

**Floating Point Data** Exponents are optional. An “E” or “e” character must be added before the exponent.

The selection of “E” or “e” is optional. “D” or “d” can not be used.

**!!, # Comment Line** Lines starting with “!!” or “#” are considered to be comment lines, and are disregarded.

A comment line can be inserted in any position in the file, and there are no restrictions on the number of lines.

### 1.6.3 Header List of Single Domain Mesh Data

The single domain mesh data consists of the following headers.

Header Name	Contents	Description No.
!AMPLITUDE	Unsteady load	M1-1
!EGROUP	Element group	M1-2
!ELEMENT	Element information	M1-3
!EQUATION	Restricted point data	M1-4
!HEADER	Title of mesh data	M1-5
!MATERIAL	Material information	M1-6
!NGROUP	Node group	M1-7
!NODE	Node information	M1-8

Header Name	Contents	Description No.
!SECTION	Section information	M1-9
!SGROUP	Surface group	M1-10
!ZERO	Absolute zero-point	M1-11
!CONTACT PAIR	Contact surface pair	M1-12
!END	Read end	M1-13

In each header, there are data items which are compatible to the parameter and each header.

Each of the above headers is briefly described in the following with examples of data creation. The number indicated on the right end of the data creation is the description number of the above Table.

### 1.6.3.1 Example of Mesh Data

```
!HEADER M1-5
TEST MODEL CTLR10

!NODE M1-8
1, 0.00000E+00, 0.00000E+00, 0.00000E+00
2, 0.50000E+01, 0.00000E+00, 0.00000E+00
3, 0.10000E+02, 0.00000E+00, 0.00000E+00
... ... ...

!ELEMENT, TYPE=351 M1-3
1, 1, 2, 4, 34, 35, 37
2, 2, 5, 4, 35, 38, 37
3, 2, 3, 5, 35, 36, 38

!SECTION, TYPE=SOLID, EGRP=ALL, MATERIAL=M1 M1-9
1.0

!MATERIAL, NAME=M1, ITEM=2 M1-6
!ITEM=1, SUBITEM=2
2.1E5, 0.3
!ITEM=2, SUBITEM=1
7.8e-6

!NGROUP, NGRP=FIX, GENERATE M1-7
2, 2, 1
3, 3, 1
1, 1, 1
69, 69, 1
67, 67, 1

!NGROUP, NGRP=CL1 M1-8
50

!END M1-13
```

### 1.6.3.2 (1) !AMPLITUDE (M1-1)

Specifies the changes of time of the variables which provide the load conditions in the step.

```
!AMPLITUDE, NAME=<name> [ , optional parameter]
VAL1, T1, VAL2, T2, VAL3, T3 ...
(up to four items in one line)
```

Parameter	
NAME	Name (mandatory)
DEFINITION	Type (omissible)
TIME	Type of time (omissible)
VALUE	Type of value (omissible)
INPUT	External file name (omissible)

Parameter Name	Parameter Value	Contents
NAME	<name>	AMPLITUDE Name
DEFINITION	TABULAR	Default (default only in current version)
TIME	STEP TIME	Default (default only in current version)
VALUE	RELATIVE	Relative value (default)
	ABSOLUTE	Absolute value
INPUT	<filename>	External file name (omissible), can also be use together with the 2nd line or later

Parameter Name	Attributions	Contents
VAL1	R	Value at time T1
T1	R	Time T1
VAL2	R	Value at time T2
T2	R	Time T2
VAL3	R	Value at time T3
T3	R	Time T3

### 1.6.3.3 (2) !EGROUP (M1-2)

Definition of element group

!EGROUP, EGRP=<egrp> [ , optional parameter ]

Parameter		
EGRP	Element group name (mandatory)	
GENERATE	Automatic generation of nodes belonging to the element group (omissible)	
INPUT	External file name (omissible)	

Parameter	Parameter Name	Contents
	Name	
EGRP	<egrp>	Element group name
GENERATE	N/A	Automatic generation of nodes belonging to the element group
INPUT	<filename>	External file name (omissible), can also be use together with the 2nd line or later

2nd Line or later (when GENERATE is not used)

(2nd Line) elem1 , elem2 , elem3 ...  
(Hereinafter the same)

Parameter Name	Attributions	Contents
elemX	I	Element number belonging to the element group

2nd Line or later (when GENERATE used)

(2nd Line) elem1 , elem2 , elem3  
(Hereinafter the same)

Parameter		
Name	Attributions	Contents
elem1	I	First element number in the element group
elem2	I	Last element number in the element group
elem3	I	Element number increment (omissible, number becomes elem3=1 when omitted)

Note:

- Any number of elements can be inserted in one line. Any number of lines can be inserted until the next option starts.
- It is necessary to define the element to be specified before “!EGROUP”.
- The element not defined in the “!ELEMENT” option will be excluded, and a warning message will be displayed.
- When the specified element exists in the same group, it will be ignored and a warning message will be displayed. All the elements belong to the element group named “ALL” (generated automatically). One group can be defined by dividing into multiple groups.

Example of Use

```
!EGROUP, EGRP=EA01
 1, 2, 3, 4, 5, 6
 101, 102
 205
!EGROUP, EGRP=EA02
 101, 102
!EGROUP, EGRP=EA01          "501, 505" are added to group "EA01".
 501, 505
!EGROUP, EGRP=EA04, GENERATE      "301, 303, 305, 307, 309, 311, 312, 313"
 301, 309, 2                  are added to group "NA04".
 311, 313
```

#### 1.6.3.4 !ELEMENT (M1-3)

Definition of elements

1st Line

**!ELEMENT**, TYPE=<type> [ , optional parameter ]

Parameter	
TYPE	Element type (mandatory)
EGRP	Element group name (omissible)
MATITEM	Number of physical property items when defining the material physical properties for each element (not used when defining physical properties for each section)
INPUT	External file name (omissible)

Parameter	Parameter	
Name	Value	Contents
TYPE	111	Rod, link element (Linear)
	231	Triangular element (Linear)
	232	Triangular element (Quadratic)

Parameter Name	Parameter Value	Contents
241		Quadrilateral element (Linear)
242		Quadrilateral element (Quadratic)
301		Truss element (Linear)
341		Tetrahedral element (Linear)
342		Tetrahedral element (Quadratic)
351		Triangular prism element (Linear)
352		Triangular prism element (Quadratic)
361		Hexahedral element (Linear)
362		Hexahedral element (Quadratic)
541		Interface element (Quadrilateral cross section, Linear)
611		Beam element (Linear)
641		Beam element (Linear, with 3-dof nodes)
731		Triangular shell element (Linear)
741		Quadrilateral shell element (Linear)
743		Quadrilateral shell element (Quadratic)
761		Triangular shell element (Linear, with 3-dof nodes)
781		Quadrilateral shell element (Linear, with 3-dof nodes)
EGRP	<egrp>	Element group name (omissible)
INPUT	<filename>	External file name (omissible), can also be used together with the 2nd line or later

2nd Line or later

(2nd Line) ELEM\_ID, nod1, nod2, nod3, ..., MAT1, MAT2, ...  
 (Hereinafter the same)

Parameter Name	Attributions	Contents
ELEM_ID	I	Element number
nodX	I	Connectivity
MATy	R	Physical Property value for each element

Note:

- For details of the element types and connectivity, refer to “Chapter 4 Element Library”.
- The node specified by the connectivity must be defined before “!ELEMENT”.
- The element numbers do not have to be continued.
- The “!ELEMENT” option can be defined any number of times.
- The element number must be a natural number. This can not be omitted.
- When the same element number is used repeatedly, the value input last will be used. In this case, a warning message will be output.
- Undefined nodes can not be used for connectivity.
- The definition of one element can be described in multiple lines.

Example of Use

```
!ELEMENT, TYPE=231
 1, 1, 2, 3
 2, 4, 8, 5
 4, 6, 7, 8
!ELEMENT, TYPE=361, EGRP=A
 101, 101, 102, 122, 121, 201, 202, 222, 221
 102, 102, 103, 123, 122, 202, 203, 223, 222
 103, 103, 104, 124, 123, 203, 204, 224, 223
```

### 1.6.3.5 (4) !EQUATION (1-4)

Definition of restricted node group

1st Line

**!EQUATION [ , optional parameter ]**

Parameter	
INPUT	External file name (omissible)

Parameter Name	Parameter Value	Contents
INPUT	<filename>	External file name (omissible), can also be used together with the 2nd line or later

2nd Line or later

(2nd Line) NEQ, CONST

(3rd Line or later) nod1, DOF1, A1, nod2, DOF2, A2 ... (up to seven terms for one line)  
(Hereinafter repeated)

Parameter Name	Attributions	Contents
NEQ	I	Number of equation terms
CONST	R	Constant term of equation (right value)
nod1	I/C	1st node or node group
DOF1	I	Restricted degree of freedom of 1st node or node group
A1	R	Factor of 1st node or node group
nod2	I/C	2nd node or node group
DOF2	I	Restricted degree of freedom of 2nd node or node group
A2	R	Factor of 2nd node or node group

Note:

- When a node or a node group not defined by “!NODE” is specified, it will be ignored and a warning message will be displayed.
- In the case of “nod1=nod2”, it will be ignored and a warning message will be displayed.
- When a node group is specified, if the number of nodes is not consistent an error will occur.
- The degree of freedom number differs by the type of analysis and elements. An inconsistent degree of freedom will be ignored, and a warning message will be displayed.

Example of Use

```
!EQUATION
3
101, 1, 1.0, 102, 1, -1.0, 103, 1, -1.0
2
NG1, 2, 1.0, NG5, 2, -1.0
```

### 1.6.3.6 (3) !HEADER (M1-5)

Title of mesh data

1st Line

**!HEADER**

Parameter
N/A

2nd Line or later

(2nd Line) TITLE

Parameter Name	Attributions	Contents
TITLE	C	Header title

Example of Use

!HEADER

Mesh for CFD Analysis

Note:

- Omissible
- Although the header can use multiple lines, it can be recognized as a header up to the 127th column of the first line.
- When “!HEADER” is defined multiple times, the contents will be updated and a warning message will be displayed.

#### 1.6.3.7 (6) !MATERIAL (M1-6)

Definition of material physical properties.

When the physical properties depend on the temperature, table input can be performed for each compatible temperature.

The table input can also be performed for the relationship of the stress strain in a stress analysis.

1st Line

!MATERIAL, NAME=<name> [ , optional parameter ]

Parameter	
NAME	Material name (mandatory)
ITEM	Number of physical property items (omissible, becomes “1” when omitted)
INPUT	External file name (omissible)

Parameter Name	Parameter Value	Contents
NAME	<name>	Material name
ITEM	<ITEMnum>	Number of physical property items by user definition
INPUT	<filename>	External file name (omissible), can also be used together with the 2nd line or later

2nd Line or later

(2nd Line) !ITEM=1, SUBITEM=<k>

(3rd Line) VAL1-1-1, VAL1-1-2, ... VAL1-1-k, TEMP1-1

(4th Line) VAL1-2-1,VAL1-2-2, ... VAL1-2-k, TEMP1-2

(L+2nd Line) VAL1-L-1,VAL1-L-2, ... VAL1-L-k, TEMP1-L

Hereinafter, the definition is repeated until “!ITEM=<ITEMnum>”.

---

Subparameter (for “!ITEM”)

SUBITEM	Number of sub physical property items defined in each of the physical property items(omissible, becomes “1” when omitted, used to define anisotropy and etc.)
---------	---

---

Subparameter

Name	Parameter Value	Contents
SUBITEM	<subITEMnum>	Number of sub physical property items by user definition

---

[When the m-th physical property depends on the temperature]

When the number of items of the temperature dependent table is N, input as follows:

```
!ITEM=m, SUBITEM=k
VALm1-1, ..., VALm1-k, TEMPm1
VALm2-1, ..., VALm2-k, TEMPm2
...
VALmN-1, ..., VALmN-k, TEMPm-N
```

Parameter Name	Attributions	Contents
VALmn-k	R	Physical property value (Temperature dependent)
TEMPmn	R	Compatible temperature

---

Must be input as TEMPm1 < TEMPm2 < ... < TEMPmN.

VALm1 is used when the temperature is TEMPm1 or below, and VALmN is used when the temperature is TEMPmN or more.

[When the m-th physical property does not depend on the temperature]

```
!ITEM=m, SUBITEM=k
VALm1-1, ..., VALm1-k
VALm2-1, ..., VALm2-k
...
VALmN-1, ..., VALmN-k
```

Parameter Name	Attributions	Contents
VALmn-k	R	Physical property value (not temperature dependent)

---

Note:

- When the material name is duplicated, an error will occur.
- When the MATERIAL referred to in the “!SECTION” option is not defined, an error will occur.
- The value used to input the physical property for each element using the parameter “MATITEM” in the “!ELEMENT” option, is used preferentially. In this case, the physical property value input using the “!MATERIAL” option will not be used.
- When the number of “!ITEM=m” suboptions and the number of parameters “ITEM” is not consistent, or when there is an undefined suboption, an error will occur.
- The “!ITEM=m” suboption does not have to be in order from the smaller m.
- When using the “!SUBITEM=k” suboption and the temperature dependency, the omitted value will become “0.0”.
- When using the temperature dependency, it must be defined in order from the lower temperature.
- When using the temperature dependency, if the same temperature is used twice or more, an error will occur.

## Example of Use

```

!MATERIAL, NAME= STEEL, ITEM= 2
!ITEM=1                               ——— No temperature dependency
 35.0
!ITEM=2
 40.0, 0.0
 45.0, 100.0
 50.0, 200.0
!MATERIAL, NAME= CUPPER              ——— Number of items = 1 (Default value)
!ITEM=1                               ——— No temperature dependency
 80.0

```

## Incorrect Example of Use

- Example 1: [Number of parameter “ITEM” and “!ITEM=m” suboptions are not consistent -1]

```

!MATERIAL, NAME= STEEL, ITEM= 2
!ITEM=3
 20.0
!ITEM=1
 35.0
!ITEM= 2
 40.0

```

- Example 2: [Number of parameter “ITEM” and “!ITEM=m” suboptions are not consistent -2]

```

!MATERIAL, NAME= STEEL, ITEM= 3
!ITEM=3
 20.0
!ITEM= 2
 40.0
!MATERIAL, NAME= CUPPER
...

```

## Elastic Static Analysis and Eigenvalue Analysis

```

!MATERIAL, NAME=<name>, ITEM=<ITEMnum>
!ITEM=1, SUBITEM=2
  <Yang_modulus>, <Poisson_ratio>
!ITEM=2
  <Density>
!ITEM=3
  <Expansion_coeff>

```

Parameter Name	Parameter Value	Contents
NAME	<name>	Compatible to material name, and MATERIAL of !SECTION
ITEM	<ITEMnum>	Number of physical property items by user definition (1 or more) <Yang_modulus> ... Young's modulus (mandatory) <Poisson_ratio> ... Poisson's ratio (mandatory) <Density> ... Mass density (mandatory when ITEMnum=3) <Expansion_coeff> ... Coefficient of linear expansion (when ITEMnum=3)

## Example

```

!! Intention of defining three types of property values
!! in the material of material name M1
!MATERIAL, NAME=M1, ITEM=3

!! The Young's modulus and Poisson's ratio is defined in !ITEM=1 (mandatory)

```

```

!ITEM=1, SUBITEM=2
 4000., 0.3

!! The mass density must be defined in !ITEM=2 (mandatory in the case of ITEM=3)
!ITEM=2
 8.0102-E10

!! The coefficient of linear expansion must be defined in !ITEM=3
!ITEM=3
 1.0-E5

```

### Heat Conduction Analysis

In the case of link, plane surface, solid and three-dimensional plate elements

```

!MATERIAL, NAME=<name>, ITEM=3
!ITEM=1, SUBITEM=2
  <Density>, <Temperature>
!ITEM=2, SUBITEM=2
  <Specific_heat>, <Temperature>
!ITEM=3, SUBITEM=2
  <Conductivity>, <Temperature>

```

Parameter Name	Parameter Value	Contents
NAME	<name>	Compatible to material name, and MATERIAL of !SECTION
ITEM	<ITEMnum>	Number of physical property items by user definition (always 3)<Density> ... Density<Specific_heat> ... Specific heat<Conductivity> ... Thermal conductivity<Temperature> ... Temperature

### Example

```

!! Intention of defining three types of property
!! values in the material of material name M1
!MATERIAL, NAME=M1, ITEM=3

!! The density and temperature are defined in !ITEM=1 (mandatory)
!ITEM=1, SUBITEM=1
 7850., 300.
 7790., 500.
 7700., 800.

!! The specific heat and temperature are defined in !ITEM=2 (mandatory)
!ITEM=2, SUBITEM=1
 0.465, 300.
 0.528, 500.
 0.622, 800.

!! The thermal conductivity and temperature are defined in !ITEM=3 (mandatory)
!ITEM=3
 43., 300.
 38.6, 500.
 27.7, 800.

```

In the case of interface element

Defined in the !SECTION header. (Material data is not required)

### Example

```
!! Definition of section
!SECTION, TYPE=INTERFACE, EGRP=GAP
 1.0, 20.15, 8.99835E-9, 8.99835E-9
```

In the above !SECTION, the gap parameter of the element belonging to the “group name = GAP” in the interface element is defined.

- 1st parameter : Gap width
- 2nd parameter : Gap heat transfer coefficient
- 3rd parameter : Gap radiation factor 1
- 4th parameter : Gap radiation factor 2

Reference

```
program TEST
use hecmw
implicit REAL*8 (A-H,O-Z)
type (hecmwT_local_mesh) :: hecMESH

!C
!C !MATERIAL, NAME=SUS304, ITEM=3
!C !ITEM=1, SUBITEM= 3
!C     100.0, 200.0, 300.0, 0.00
!C     101.0, 210.0, 301.0, 1.00
!C     102.0, 220.0, 302.0, 2.00
!C     103.0, 230.0, 303.0, 3.00
!C !ITEM=3, SUBITEM= 2
!C     1000.0, , 0.00
!C     1001.0, 1., 1.00
!C     1002.0, 2., 2.00
!C     1003.0, 3., 3.00
!C !ITEM=2
!C     5000.0
!C
!C !MATERIAL, NAME=FEC, ITEM=2
!C !ITEM=1, SUBITEM= 3
!C     2100.0, 2200.0, 2300.0, 0.00
!C     2101.0, 2210.0, 2301.0, 1.00
!C     2102.0, 2220.0, 2302.0, 2.00
!C     2103.0, 2230.0, 2303.0, 3.00
!C     3103.0, 3230.0, 2304.0, 4.00
!C !ITEM=2
!C     6000.0, 10.0
!C     6500.0, 30.0
!C

hecMESH%material%n_mat = 2

nn= hecMESH%material%n_mat
allocate (hecMESH%material%mat_name(nn))

hecMESH%material%mat_name(1)= 'SUS304'
hecMESH%material%mat_name(2)= 'FEC'

nn= hecMESH%material%n_mat
allocate (hecMESH%material%mat_ITEM_index(0:nn))
hecMESH%material%mat_ITEM_index(0)= 0
hecMESH%material%mat_ITEM_index(1)= 3
hecMESH%material%mat_ITEM_index(2)= hecMESH%material%mat_ITEM_index(1) + 2
```

```

hecMESH%material%n_mat_ITEM= hecMESH%material%mat_ITEM_index(hecMESH%material%n_mat)

nn= hecMESH%material%n_mat_ITEM
allocate (hecMESH%material%mat_subITEM_index(0:nn))

hecMESH%material%mat_subITEM_index(0)= 0
hecMESH%material%mat_subITEM_index(1)= 3
hecMESH%material%mat_subITEM_index(2)= hecMESH%material%mat_subITEM_index(1) + 1
hecMESH%material%mat_subITEM_index(3)= hecMESH%material%mat_subITEM_index(2) + 2
hecMESH%material%mat_subITEM_index(4)= hecMESH%material%mat_subITEM_index(3) + 3
hecMESH%material%mat_subITEM_index(5)= hecMESH%material%mat_subITEM_index(4) + 1

hecMESH%material%n_mat_subITEM=
&           hecMESH%material%mat_subITEM_index(hecMESH%material%n_mat_ITEM)

nn= hecMESH%material%n_mat_subITEM
allocate (hecMESH%material%mat_TABLE_index(0:nn))
hecMESH%material%mat_TABLE_index( 0)= 0
hecMESH%material%mat_TABLE_index( 1)= 4
hecMESH%material%mat_TABLE_index( 2)= hecMESH%material%mat_TABLE_index( 1) + 4
hecMESH%material%mat_TABLE_index( 3)= hecMESH%material%mat_TABLE_index( 2) + 4
hecMESH%material%mat_TABLE_index( 4)= hecMESH%material%mat_TABLE_index( 3) + 1
hecMESH%material%mat_TABLE_index( 5)= hecMESH%material%mat_TABLE_index( 4) + 4
hecMESH%material%mat_TABLE_index( 6)= hecMESH%material%mat_TABLE_index( 5) + 4
hecMESH%material%mat_TABLE_index( 7)= hecMESH%material%mat_TABLE_index( 6) + 5
hecMESH%material%mat_TABLE_index( 8)= hecMESH%material%mat_TABLE_index( 7) + 5
hecMESH%material%mat_TABLE_index( 9)= hecMESH%material%mat_TABLE_index( 8) + 5
hecMESH%material%mat_TABLE_index(10)= hecMESH%material%mat_TABLE_index( 9) + 2

hecMESH%material%n_mat_TABLE=
&           hecMESH%material%mat_TABLE_index(hecMESH%material%n_mat_subITEM)

nn= hecMESH%material%n_mat_TABLE
allocate (hecMESH%material%mat_VAL( nn))
allocate (hecMESH%material%mat_TEMP(nn))

hecMESH%material%mat_VAL = 0.d0
hecMESH%material%mat_TEMP= 0.d0

hecMESH%material%mat_VAL( 1)= 100.0d0
hecMESH%material%mat_TEMP( 1)= 0.0d0
hecMESH%material%mat_VAL( 2)= 101.0d0
hecMESH%material%mat_TEMP( 2)= 1.0d0
hecMESH%material%mat_VAL( 3)= 102.0d0
hecMESH%material%mat_TEMP( 3)= 2.0d0
hecMESH%material%mat_VAL( 4)= 103.0d0
hecMESH%material%mat_TEMP( 4)= 3.0d0

hecMESH%material%mat_VAL( 5)= 200.0d0
hecMESH%material%mat_TEMP( 5)= 0.0d0

hecMESH%material%mat_VAL(13)= 5000.0d0

hecMESH%material%mat_VAL(14)= 1000.0d0
hecMESH%material%mat_TEMP(14)= 0.0d0
hecMESH%material%mat_VAL(15)= 1001.0d0
hecMESH%material%mat_TEMP(15)= 1.0d0
hecMESH%material%mat_VAL(16)= 1002.0d0
hecMESH%material%mat_TEMP(16)= 2.0d0

```

```

hecMESH%material%mat_VAL (17)= 1003.0d0
hecMESH%material%mat_TEMP (17)= 3.0d0

hecMESH%material%mat_VAL (18)= 0.0d0
hecMESH%material%mat_TEMP (18)= 0.0d0
hecMESH%material%mat_VAL (19)= 1.0d0
hecMESH%material%mat_TEMP (19)= 1.0d0
hecMESH%material%mat_VAL (20)= 2.0d0
hecMESH%material%mat_TEMP (20)= 2.0d0
hecMESH%material%mat_VAL (21)= 3.0d0
hecMESH%material%mat_TEMP (21)= 3.0d0

hecMESH%material%mat_VAL (22)= 2100.0d0
hecMESH%material%mat_TEMP (22)= 0.0d0
hecMESH%material%mat_VAL (23)= 2101.0d0
hecMESH%material%mat_TEMP (23)= 1.0d0
hecMESH%material%mat_VAL (24)= 2102.0d0
hecMESH%material%mat_TEMP (24)= 2.0d0
hecMESH%material%mat_VAL (25)= 2103.0d0
hecMESH%material%mat_TEMP (25)= 3.0d0
hecMESH%material%mat_VAL (26)= 3103.0d0
hecMESH%material%mat_TEMP (26)= 4.0d0

write (*, '(a,i10)') '%n_mat_ITEM ', hecMESH%material%n_mat_ITEM
write (*, '(a,i10)') '%n_mat_subITEM', hecMESH%material%n_mat_subITEM
write (*, '(a,i10)') '%n_mat_TABLE ', hecMESH%material%n_mat_TABLE

end program TEST

```

### 1.6.3.8 (7) !NGROUP (M1-7)

Definition of node group

1st Line

!NGROUP, NGRP=<ngrp> [, optional parameter]

---

Parameter

---

NGRP	Node group name (mandatory)
GENERATE	Automatic generation of nodes belonging to the node group (omissible)
INPUT	External file name (omissible)

---



---

Parameter Name	Parameter Value	Contents
NGRP	<ngrp>	Node group name GENERATE N/A Automatic generation of nodes belonging to the node group
INPUT	<filename>	External file name (omissible), can also be used together with the 2nd line or later

---

2nd Line or later (when GENERATE is not used)

(2nd Line) nod1, nod2, nod3  
 (Hereinafter the same)

Parameter Name	Attributions	Contents
nodX	I	Node number belonging to the node group

2nd Line or later (when GENERATE is used)

(2nd Line) nod1, nod2, nod3  
 (Hereinafter the same)

Parameter Name	Attributions	Contents
nod1	I	First node number in the node group
nod2	I	Last node number in the node group
nod3	I	Node number increment (omissible, number becomes nod3=1 when omitted)

Note:

- Any number of nodes can be inserted in one line. Any number of lines can be inserted until the next option starts.
- It is necessary to define the nodes to be specified before “!NGROUP”.
- The node not defined in the “!NODE” option will be excluded, and a warning message will be displayed.
- When the specified node exists in the same group, it will be ignored and a warning message will be displayed.
- All the nodes belong to the node group named “ALL” (generated automatically).
- One group can be defined by dividing into multiple groups.

Example of Use

```
!NGROUP, NGRP=NA01
  1, 2, 3, 4, 5, 6
  101, 102
!NGROUP, NGRP=NA02
  101, 102
!NGROUP, NGRP=NA01
  501, 505
      ----- "501 and 505" are added to group "NA01".
!NGROUP, NGRP=NA02
  501, 505
      ----- "501 and 505" are added to group "NA02".
!NGROUP, NGRP=NA04,GENERATE
  301, 309, 2
      ----- "301, 303, 305, 307, 309, 311, 312, 313"
      are added to group "NA04".
  311, 313
```

### 1.6.3.9 (8) !NODE (M1-8)

Definition of node coordinates

1st Line

!NODE [, optional parameter]

Parameter	
SYSTEM	Coordinate system (omissible)
NGRP	Node group name (omissible)
INPUT	External file name (omissible)

Parameter Name	Parameter Value	Contents
SYSTEM	R C	Cartesian coordinate system (Default value) Cylindrical coordinate system
NGRP	<ngrp>	Node group name (omissible)
INPUT	<filename>	External file name (omissible), can also be used together with the 2nd line or later

2nd Line or later

(2nd Line) NODE\_ID, Xcoord, Ycoord, Zcoord  
(Hereinafter the same)

Parameter Name	Attributions	Contents
NODE_ID	I	Node number
Xcoord	R	X coordinate
Ycoord	R	Y coordinate
Zcoord	R	Z coordinate

Note:

- When node coordinates including the punctuation mark is omitted, the value will become “0.0”.
- When an already defined node is redefined, the contents will be updated and a warning message will be displayed.
- The node which is not referred to in “!ELEMENT” will be excluded.
- The node defined in “!ELEMENT” must be defined before “!ELEMENT”.

Example of Use

```
!NODE, NGRP=TEST
1, 0.0, 0.0, 0.5
2, 0.0, 0.0, 1.0
3, 0.0,,1.5           ----- Y coordinate is "0.0"
4,                   ----- X, Y and Z coordinates are "0.0"
```

### 1.6.3.10 (9) !SECTION (M1-9)

Definition of section

1st Line

!SECTION, TYPE=<type>, EGRP=<egrp> [ , optional parameter ]

Parameter	
TYPE	Section type (mandatory)
EGRP	Element group name (mandatory)
MATERIAL	User defined material name (mandatory)
SECOPT	Auxiliary parameter for the element type (omissible, becomes =0 when omitted)
INPUT	External file name (omissible)

Parameter Name	Parameter Value	Contents
TYPE	SOLID	Rod, triangular, quadrilateral, tetrahedral, pentahedral, hexadedral elements
	SHELL	Shell element
	BEAM	Beam element
	INTERFACE	Interface element

Parameter Name	Parameter Value	Contents
EGRP	<egrp>	Element group name
MATERIAL	<material>	Material name defined by user
SECOPT	<secopt>	= 0: Not specified, plane stress = 1: Plane strain = 2: Axial symmetry = 10: 0 + reduced integration = 11: 1 + reduced integration = 12: 2 + reduced integration
INPUT	<filename>	External file name (omissible), can also be used together with the 2nd line or later

2nd Line or later

#### In the case of [TYPE=SOLID]

(2nd Line) THICKNESS

Parameter Name	Attributions	Contents
THICKNESS	R	Element thickness, cross-sectional area

In the case of “TYPE=SOLID”, the “THICKNESS” can be omitted, and default value (1.0) is inserted.

#### In the case of [TYPE=SHELL]

(2nd Line) THICKNESS, INTEGPOINTS

Parameter Name	Attributions	Contents
THICKNESS	R	Shell cross section thickness
INTEGPOINTS	I	Integral point in shell cross sectional direction

#### In the case of [TYPE=BEAM]

(2nd Line) vx , vy , vz , area , Iyy , Iz , Jx

Parameter Name	Attributions	Contents
vx,vy,vz	R	Direction cosine of reference axis
area	R	Area of cross section
Iyy, Izz	R	Second moment of cross section
Jx	R	Torsion constant of cross section

#### In the case of [TYPE=INTERFACE]

(2nd Line) THICKNESS, GAPCON, GAPRAD1, GAPRAD2

Parameter Name	Attributions	Contents
THICKNESS	R	Cross-sectional thickness
GAPCON	R	Gap heat transfer coefficient (0 when omitted)
GAPRAD1	R	Gap radiant heat transfer factor-1 (0 when omitted)
GAPRAD2	R	Gap radiant heat transfer factor-2 (0 when omitted)

Note:

- When the parameter “TYPE” is not consistent with the element type, an error will occur.
- When there is an element without SECTION information, an error will occur.
- When the section name is duplicated, an error will occur.

Example of Use

```
!SECTION, EGRP=SOLID1, TYPE=SOLID, MATERIAL=STEEL  
!SECTION, EGRP=SHELL2, TYPE=SHELL, MATERIAL=STEEL  
1.0, 5
```

### 1.6.3.11 (10) !SGROUP (M1-10)

Definition of surface group

1st Line

```
!SGROUP, SGRP=<sgrp> [, optional parameter]
```

Parameter	
SGRP	Surface group name (mandatory)
INPUT	External file name (omissible)

Parameter Name	Parameter Value	Contents
SGRP	<sgrp>	Surface group name
INPUT	<filename>	External file name (omissible), can also be used together with the 2nd line or later

2nd Line or later

```
(2nd Line) elem1, lsuf1, elem2, lsuf2, elem3, lsuf3, ...  
(Hereinafter the same)
```

Parameter Name	Attributions	Contents
elemX	I	Element number belonging to the surface group
lsufX	I	Local surface number of the element belonging to the surface group

Note:

- For the element type and surface number, refer to “Chapter 4 Element Library”.
- The surface consists of a combination of (elements and local surface numbers). Any number of surfaces can be inserted in one line. Any number of lines can be inserted until the next option starts. The combination of (elements and local surface numbers) must be in the same line.
- It is necessary to define the element to be specified before “!SGROUP”.
- The element not defined in “!ELEMENT” option will be excluded, and a warning message will be displayed.
- The surface which includes the element not defined in “!ELEMENT” option will be excluded, and a warning message will be displayed.
- The surface where the element type and the surface number are not consistent will be excluded, and a warning message will be displayed.
- One group can be defined by dividing into multiple groups.

Example of Use

```
!SGROUP, SGRP= SUF01
```

```

101, 1, 102, 1, 103, 2, 104, 2
201, 1, 202, 1
501, 1
!SGROUP, SGRP= SUF02
101, 2, 102, 2
!SGROUP, SGRP= EA01      "(601,1) and (602 2)" are added to group "SUF01".
601, 1
602, 2

```

#### Incorrect Example of Use

- Example 1: [When (elements, and local surface numbers) group exists in multiple lines]

```

!SGROUP, SGRP= SUF01
101, 1, 102, 1, 103
1, 104, 1

```

- Example 2: [Local surface numbers and element type are not consistent]

```

!ELEMENT, TYPE= 211, SECTION= A
101, 1, 2, 3
102, 2, 3, 4
...
!SGROUP, SGRP= SUF01
101, 1
101, 2
101, 4      Since a 4th surface does not exist in a triangular element,
             this combination will be disregarded.

```

#### 1.6.3.12 (11) !ZERO (M1-11)

Absolute zero-point

1st Line

```
!ZERO
```

Parameter
N/A

2nd Line or later

```
(2nd Line) ZERO
```

Parameter Name	Attributions	Contents
ZERO	R	Absolute zero-point

Note:

- Omissible. Becomes “absolute zero-point = 0” when omitted.
- When “!ZERO” is defined multiple times, the contents will be updated and a warning message will be displayed.

Example of Use

```
!ZERO
-273.16
```

#### 1.6.3.13 (12) !CONTACT PAIR (M1-12)

Definition of contact surface pair used for contact analysis

1st Line

**!CONTACT PAIR, NAME=<name> [ , optional parameter ]**

Parameter	
NAME	Contact pair name (mandatory)
TYPE	Type (omissible)

Parameter Name	Parameter Value	Contents
TYPE	NODE-SURF SURF-SURF	Slave surface is the node group Master surface is the surface group (default) Both the slave surface and master surface are the surface group

2nd Line or later

(2nd Line or later) SLAVE\_GRP, MASTER\_GRP  
(Hereinafter the same)

Parameter Name	Attributions	Contents
SLAVE_GRP	C	Slave surface node / surface group name
MASTER_GRP	C	Surface group name of the master surface

Note:

- When using mesh-refinement functionality, always use “TYPE=SURF–SURF”.
- Even when “TYPE=SURF–SURF” is used, the internal contact algorithm within FrontISTR is one for point-to-surface type contact.

Reference

- When “TYPE=SURF–SURF” is used, the contact pair is converted to a node-surface type contact pair within FrontISTR by automatically generating a node-group from the slave surface-group.
- The group-name of the automatically generated node group is concatenation of “FSTR\_S2N\_” and the group-name of the original slave surface-group.

#### 1.6.3.14 (13) !END (M1-13)

End of mesh data

When this header is displayed, the reading of the mesh data is completed.

#### 1.6.3.15 1st Line

**!END**

Parameter
N/A

#### 1.6.3.16 2nd Line or later

Parameter
N/A

## 1.7 Analysis Control Data

### 1.7.1 Outline of Analysis Control Data

In FrontISTR, an analysis control data file is input to acquire the computing control data, solver control data and post process (visualization) control data as shown in the following figure, in order to implement the analytical calculations.

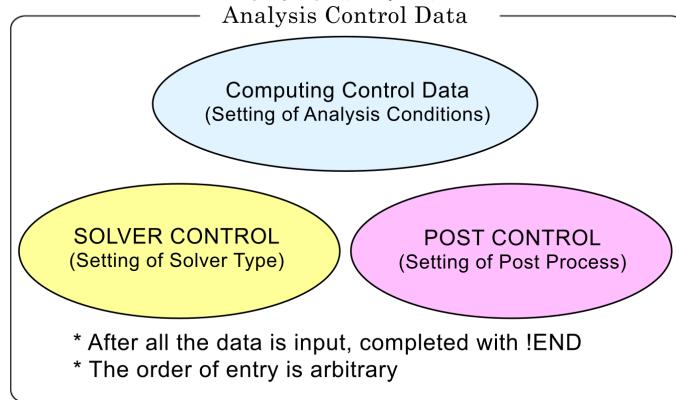


Figure 17: Analysis Control Data

The features of the analysis control data file are as follows.

- This is an ASCII format file based on a free format.
- This file consists of a header which starts with “!” and the data following this.
- The order of description of the header is basically free.
- A “,” is used as a punctuation mark of the data.
- The inside of the file is briefly divided into three zones.
- “!END” is input at the end of the file for completion.

### Example of Analysis Control Data

```

#####
##### (1) Computing control data portion #####
#####
### Control File for HEAT solver
!SOLUTION,TYPE=HEAT
!FIXTEMP
XMIN, 0.0
XMAX, 500.0

#####
##### (2) Solver control data portion #####
#####
### Solver Control
!SOLVER,METHOD=1,PRECOND=1,ITERLOG=NO,TIMELOG=NO
100, 1
1.0e-8,1.0,0.0

#####
##### (3) Post control (visualization) data portion #####
#####
### Post Control
!WRITE,RESULT
!WRITE,VISUAL

```

```

!VISUAL, method=PSR
!surface_num = 1
!surface 1
!surface_style = 1
!display_method 1
!color_comp_name = TEMPERATURE
!color_subcomp = 1
!output_type = BMP
!x_resolution = 500
!y_resolution = 500
!num_of_lights = 1
!position_of_lights =
-20.0, 5.8, 80.0
!viewpoint = -20.0 10.0 8.0
!up_direction = 0.0 0.0 1.0
!ambient_coef= 0.3
!diffuse_coef= 0.7
!specular_coef= 0.5
!color_mapping_style= 1
!!interval_mapping= -0.01, 0.02
!color_mapping_bar_on = 1
!scale_marking_on = 1
!num_of_scale = 5
!font_size = 1.5
!font_color = 1.0 1.0 1.0
!END

```

### 1.7.2 Input Rules

The analysis control data consists of a header line, data line and a comment line.

One header is always included in the header line.

**Header** The header specifies the meaning of the data and the data block in the analysis control data. When the head of the term starts with a “!”, it is considered to be a header.

**Header Line** The header and the parameter accompanying this are described in this line.

The header line must start with a header. When a parameter is required, a “,” must be used to continue after that. When the parameter takes on a value, use an “=” after the parameter and describe the value after that. The header line can not be described in more than two lines.

**Data Line** The data line starts after the header line, and the necessary data is described.

The data lines may be in multiple lines; however, this is determined according to the rules of the data description defined by each header.

There are cases where data lines are not required.

**Punctuation** A comma “,” is used as a punctuation of the data.

**Handling of Blanks** Blanks are disregarded.

**Name** Regarding the characters which can be used for the name, there is the underscore “\_”, hyphen “-”, and alphanumeric characters “a – z, A – Z, 0 – 9”; however, the first letter of the name must start with “\_”, or an alphabetic character “a – z, A – Z”. There is no distinction between uppercase and lowercase letters, and all letters are internally handled as uppercase letters.

The maximum length of the name is 63 characters.

**File Name** Regarding the characters which can be used for the file name, there are the underscore “\_”, hyphen “-”, period “.”, slash “/”, and the alphanumeric characters “a – z, A – Z, 0 – 9”.

As long as there is no specific description, a path can be included in the file name. Both the relative path and the absolute path can be specified.

The maximum length of the file name is 1,023 characters.

**Floating Point Data** Exponents are optional. An “E” or “e” character must be added before the exponent. The selection of “E” or “e” is optional.

**!!, #, Comment Line** Lines starting with “!!” or “#” are considered to be comment lines, and are disregarded. A comment line can be inserted in any position in the file, and there are no restrictions on the number of lines.

**!END** End of mesh data

When this header is displayed, the reading of the mesh data is completed.

### 1.7.3 Analysis Control Data

#### 1.7.3.1 Header List of Computing Control Data

In FrontISTR, the following items can be mentioned as the boundary conditions which can be used for the computing control data.

- Distributed load conditions (body force, pressure loading, gravity, centrifugal force)
- Concentrated load conditions
- Heat load
- Single point restriction conditions (SPC conditions)
- Spring boundary conditions
- Contact
- Concentrated heat flux
- Distributed heat flux
- Convective heat transfer boundary
- Radiant heat transfer boundary
- Specified temperature boundary

The same as the mesh data, the !HEADER format is used as the definition method of the above boundary conditions.

The header list of the common control data is shown in the following Table 7.3.1, and the header list for each analysis type is shown in Table 7.3.2.

**Table 7.3.1: Control Data Common to All Analysis**

Header	Meaning	Remarks	Description No.
!VERSION	Solver version number		1-1
!SOLUTION	Specification of analysis type	Mandatory	1-2
!WRITE,VISUAL	Specification of visualization output		1-3
!WRITE,RESULT	Specification of results output		1-4
!WRITE,LOG	Specification of results output		1-5
!OUTPUT_VIS	Control of visualization output items		1-6
!OUTPUT_RES	Control of results output items		1-7
!RESTART	Control of restarting		1-8
!ECHO	Echo output		1-9
!ORIENTATION	Definition of local coordinate system		1-10
!SECTION	Definition of local coordinate system the section correspondent to		1-11
!INITIAL_CONDITION	Definition of initial condition		1-12
!END	Ending specification of control data		1-13

**Table 7.3.2: Control Data for Static Analysis**

Header	Meaning	Remarks	Description No.
!STATIC	Static analysis control		2-1
!MATERIAL	Material name		2-2
!ELASTIC	Elastic material physical properties		2-2-1
!PLASTIC	Plastic material physical properties		2-2-2
!HYPERELASTIC	Hyperelastic material physical properties		2-2-3
!VISCOELASTIC	Viscoelastic material physical properties		2-2-4
!CREEP	Creep material physical properties		2-2-5
!DENSITY	Mass density		2-2-6
!EXPANSION_COEFF	Coefficient of linear expansion		2-2-7
!TRS	Tempearture dependent behaviour of viscoelastic material		2-2-8

Header	Meaning	Remarks	Description No.
!FLUID	Flow Condition		2-2-9
!USE_MATERIAL	User defined material		2-2-10
!BOUNDARY	Displacement boundary conditions		2-3
!SPRING	Spring boundary conditions		2-3-1
!CLOAD	Concentrated load		2-4
!DLOAD	Distributed load		2-5
!ULOAD	User defined external load		2-6
!CONTACT_ALGO	Contact analytic algorithm		2-7
!CONTACT	Contact		2-8
!TEMPERATURE	Nodal temperature in thermal stress analysis		2-9
!REFTEMP	Reference temperature in thermal stress analysis		2-10
!STEP	Analysis step control		2-11

**Table 7.3.3: Control Data for Eigenvalue Analysis**

Header	Meaning	Remarks	Description No.
!EIGEN	Eigenvalue analysis control	Mandatory in eigenvalue analysis	3-1

**Table 7.3.4: Control Data for Heat Conduction Analysis**

Header	Meaning	Remarks	Description No.
!HEAT	Heat conduction analysis control	Mandatory in heat conduction analysis	4-1
!FIXTEMP	Nodal temperature		4-2
!CFLUX	Concentrated heat flux given to node		4-3
!DFLUX	Distributed heat flux / internal heat generation given to element surface		4-4
!SFLUX	Distributed heat flux by surface group		4-5
!FILM	Heat transfer coefficient given to boundary plain		4-6
!SFILM	Heat transfer coefficient by surface group		4-7
!RADIATE	Radiation factor given to boundary plane		4-8
!SRADIATE	Radiation factor by surface group		4-9
!WELD_LINE	Weld line		4-10

**Table 7.3.5: Control Data for Dynamic Analysis**

Header	Meaning	Remarks	Description No.
!DYNAMIC	Dynamic analysis control	Mandatory in dynamic analysis	5-1
!VELOCITY	Velocity boundary conditions		5-2
!ACCELERATION	Acceleration boundary conditions		5-3
!COUPLE	Coupled surface definition Required in coupled analysis		5-4
!EIGENREAD	Specification of eigenvalues and eigenvectors	Mandatory in frequency response analysis	5-5
!FLOAD	Definition of concentrated load for frequency response analysis		5-6

In each header, there are data items which comply with the parameter and each header.

Each of the above headers is described in the following with examples of data creation for each analysis type. The description number in the above Table is the number indicated on the right end of the example of the data creation.

### 1.7.3.2 (1) Control data common to all analyses

#### 1.7.3.2.1 Example of Analysis Control Data

```
### Control File for FISTR
!VERSION                                1-1
  5
!SOLUTION, TYPE=STATIC                  1-2
!WRITE, VISUAL                           1-3
!WRITE, RESULT                           1-4
!ECHO                                     1-9
!BOUNDARY
  FIX, 1, 3, 0.0
!CLOAD                                    2-4
  CL1, 3, -1.0
!END                                      1-12
```

#### 1.7.3.2.2 Description of Header

1-1 !VERSION

Refer to the solver version.

1-2 !SOLUTION, TYPE=STATIC

TYPE=analysis type

1-3 !WRITE, VISUAL

Output of data by visualizer via memory

Outputs the file just by entering

1-4 !WRITE, RESULT

Output of analysis results file

Outputs the file just by entering

1-6 !ECHO

Output of node data, element data and material data to log file

Outputs to the file just by entering

1-8 !END

Indicates the end of control data

### 1.7.3.3 (2) Static analysis control data

#### 1.7.3.3.1 Example of Static Analysis Control data

```
### Control File for FISTR
!SOLUTION, TYPE=STATIC                  1-2
!WRITE, VISUAL                           1-3
!WRITE, RESULT                           1-4
!ECHO                                     1-9
!MATERIAL, NAME=M1                     2-2
!ELASTIC, TYPE=ISOTROPIC                2-2-1
  210000.0, 0.3
```

<b>!BOUNDARY</b>	2-3
<b>FIX</b> , 1, 3, 0.0	
<b>!SPRING</b> 2-3-1	
200, 1, 0.03	
<b>!CLOAD</b> 2-4	
CL1, 3, -1.0	
<b>!DLOAD</b> 2-5	
1, P1, 1.0	
<b>!TEMPERATURE</b>	2-9
1, 10.0	
<b>!REFTEMP</b>	2-10
<b>!STEP</b> , CONVERG=1.E-5, MAXITER=30	2-11
<b>!END</b>	1-12

### 1.7.3.3.2 Description of Header

- Red figures are the values indicated in the example.
- Alphabetic characters in the 2nd line of the table express the parameter name.

#### 2-1 !STATIC

Setting of static analysis method

#### 2-2 !MATERIAL

Definition of material physical properties

NAME = name of material physical properties

#### 2-2-1 !ELASTIC, TYPE=ISOTROPIC

Definition of elastic substance

TYPE = elastic type

Young's Modulus	Poisson's Ratio
YOUNG_MODULUS	POISSON_RATIO
210000.0	0.3

#### 2-3 !BOUNDARY

Definition of displacement boundary conditions

Node ID or Node Group Name	Start No. of Restricted Degree of Freedom	End No. of Restricted Degree of Freedom	Restricted Value
NODE_ID	DOF_idS	DOF_idE	Value
FIX,	1,	3,	0.0

#### 2-3-1 !SPRING

Definition of spring boundary conditions

Node ID or Node Group Name	Restricted Degree of Freedom	Spring Constant
NODE_ID	DOF_id	Value
200,	1,	0.03

#### 2-4 !CLOAD

Definition of concentrated load

Node ID or Node Group Name	Degree of Freedom No.	Load Value
NODE_ID	DOF_id	Value
CL1,	3,	-1.0

## 2-5 !DLOAD

Definition of distributed load

Element ID or Element Group Name	Load Type No.	Load Parameter
ELEMENT_ID	LOAD_type	param
1,	P1,	1.0

## 2-9 !TEMPERATURE

Specification of nodal temperature used for thermal stress analysis

Node ID or Node Group Name	Temperature
NODE_ID	Temp_Value
1,	10

## 2-10 !REFTEMP

Definition of reference temperature in thermal stress analysis

## 2-11 !STEP

Control of nonlinear static analysis (Omissible in the case of linear analysis)

Convergence Value Judgment Threshold	No. of Sub Steps (When AMP exists, AMP has priority)	Max No. of Iterative Calculations	Time Function Name (Specified in !AMPLITUDE)
CONVERG 1.0E-05	SUBSTEPS 10	MAXITER 30	AMP

### 1.7.3.4 (3) Eigenvalue analysis control data

#### 1.7.3.4.1 Example of Eigenvalue Analysis Control Data

```
### Control File for FISTR
!SOLUTION, TYPE=EIGEN                                1-2
!WRITE, VISUAL                                         1-3
!WRITE, RESULT                                         1-4
!ECHO                                                 1-9
!EIGEN
  3, 1.0E-8, 60                                         3-1
!BOUNDARY
  FIX, 1, 2, 0.0                                         2-3
!END
```

#### 1.7.3.4.2 Description of Header

Red figures are the values indicated in the example.

3-1 !EIGEN

Parameter settings of eigenvalue analysis

No. of Eigenvalue	Allowance	Max No. of Iterations
NSET 3,	LCZTOL 1.0E-8,	LCZMAX 60

2-3 !BOUNDARY (Same items as in Static Analysis)

Definition of displacement boundary conditions

Node ID or Node Group Name	Start No. of Restricted Degree of Freedom	End No. of Restricted Degree of Freedom	Restricted Value
NODE_ID FIX,	DOF_idS 1,	DOF_idE 3,	Value 0.0

### 1.7.3.5 (4) Heat conduction analysis control data

#### 1.7.3.5.1 Example of Heat Conduction Analysis Control Data

```
### Control File for FISTR
!SOLUTION, TYPE=HEAT           1-2
!WRITE, VISUAL                 1-3
!WRITE, RESULT                  1-4
!ECHO                           1-9
!HEAT
!FIXTEMP
  XMIN, 0.0
  XMAX, 500.0
!CFLUX
  ALL, 1.0E-3
!DFLUX
  ALL, S1, 1.0
!SFLUX
  SURF, 1.0
!FILM
  FSURF, F1, 1.0, 800
!SFILM
  SFSURF, 1.0, 800.0
!RADIATE
  RSURF, R1, 1.0E-9, 800.0
!SRADIATE
  RSURF, R1, 1.0E-9, 800.0
!END                           1-12
```

#### 1.7.3.5.2 Description of Header

Red figures are the values indicated in the example.

4-1 !HEAT

Definition of control data for calculation

!HEAT (No data)	—— Steady calculation
!HEAT 0.0	—— Steady calculation
!HEAT 10.0, 3600.0	—— Fixed time increment unsteady calculation

```

!HEAT
  10.0, 3600.0, 1.0           ----- Automatic time increment unsteady calculation
!HEAT
  10.0, 3600.0, 1.0, 20.0    ----- Automatic time increment unsteady calculation

```

#### 4-2 !FIXTEMP

Node group name, or node ID and fixed temperature

#### 4-3 !CFLUX

Definition of concentrated heat flux given to node

Node Group Name or Node ID	Heat Flux Value
NODE_GRP_NAME	Value
ALL,	1.0E-3

#### 4-4 !DFLUX

Definition of distributed heat flux and internal heat generation given to surface of element

Element Group Name or Element ID	Load Type No.	Heat Flux Value
ALL,	S1,	1.0

### Load Parameter

Load Type No.	Applied Surface	Parameter
BF	Element Overall	Calorific value
S1	Surface No. 1	Heat flux value
S2	Surface No. 2	Heat flux value
S3	Surface No. 3	Heat flux value
S4	Surface No. 4	Heat flux value
S5	Surface No. 5	Heat flux value
S6	Surface No. 6	Heat flux value
S0	Shell surface	Heat flux value

#### 4-5 !SFLUX

Definition of distributed heat flux by surface group

Surface Group Name	Heat Flux Value
SURFACE_GRP_NAME	Value
SURF,	1.0

#### 4-6 !FILM

Definition of heat transfer coefficient given to boundary plane

Element Group Name or Element ID	Load Type No.	Heat Transfer Coefficient	Ambient Temperature
ELEMENT_GRP_NAME FSURF,	LOAD_type F1,	Value 1.0,	Sink 800.0

### Load Parameter

Load Type No.	Applied Surface	Parameter
F1	Surface No. 1	Heat transfer coefficient and ambient temperature
F2	Surface No. 2	Heat transfer coefficient and ambient temperature
F3	Surface No. 3	Heat transfer coefficient and ambient temperature
F4	Surface No. 4	Heat transfer coefficient and ambient temperature
F5	Surface No. 5	Heat transfer coefficient and ambient temperature
F6	Surface No. 6	Heat transfer coefficient and ambient temperature
F0	Shell surface	Heat transfer coefficient and ambient temperature

#### 4-7 !SFILM

Definition of heat transfer coefficient by surface group

Surface Group Name	Heat Transfer Rate	Ambient Temperature
SURFACE_GRP_NAME SFSURF,	Value 1.0,	Sink 800.0

#### 4-8 !RADIATE

Definition of radiation factor given to boundary plane

Element Group Name or Element ID	Load Type No.	Radiation Factor	Ambient Temperature
ELEMENT_GRP_NAME RSURF,	LOAD_type R1,	Value 1.0E-9,	Sink 800.0

### Load Parameter

Load Type No.	Applied Surface	Parameter
R1	Surface No. 1	Radiation factor and ambient temperature
R2	Surface No. 2	Radiation factor and ambient temperature
R3	Surface No. 3	Radiation factor and ambient temperature
R4	Surface No. 4	Radiation factor and ambient temperature
R5	Surface No. 5	Radiation factor and ambient temperature
R6	Surface No. 6	Radiation factor and ambient temperature
R0	Shell surface	Radiation factor and ambient temperature

#### 4-9 !SRADIATE

Definition of radiation factor by surface group

Surface Group Name	Radiation Factor	Ambient Temperature
SURFACE_GRP_NAME SRSURF,	Value 1.0E-9,	Sink 800.0

### 1.7.3.6 (5) Dynamic analysis control data

#### 1.7.3.6.1 Example of Dynamic Analysis Control Data

```
### Control File for FISTR
!SOLUTION, TYPE=DYNAMIC
!DYNAMIC, TYPE=NONLINEAR
1 , 1
```

1-2

5-1

```

0.0, 1.0, 500, 1.0000e-5
0.5, 0.25
1, 1, 0.0, 0.0
100, 5, 1
0, 0, 0, 0, 0, 0
!BOUNDARY, AMP=AMP1
           2-3
  FIX, 1, 3, 0.0
!CLOAD, AMP=AMP1
           2-4
  CL1, 3, -1.0
!COUPLE, TYPE=1
           5-4
  SCOUPLE
!STEP, CONVERG=1.E-6, ITMAX=20
           2-11
!END
           1-12

```

### 1.7.3.6.2 Description Header

- Red figures are the values indicated in the example.
- Alphabetic characters in the 2nd line of the table express the parameter name.

#### 5-1 !DYNAMIC

Controlling the linear dynamic analysis

Solution of Equation of Motion	Analysis Types
idx_eqa	idx_resp
11	1

Analysis Start Time	Analysis End Time	Overall No. of STEPS	Time Increment
t_start	t_end	n_step	t_delta
0.0	1.0	500	1.0000e-5

Parameter $\gamma$ of Newmark- $\beta$ Method	Parameter $\beta$ of Newmark- $\beta$ Method
gamma	beta
0.5	0.25

Type of Mass Matrix	Type of Damping	Parameter $R_m$ of Rayleigh Damping	Parameter of $R_k$ of Rayleigh Damping
idx_mass	idx_dmp	ray_m	ray_k
1	1	0.0	0.0

Resules Output Interval	Monitoring Node ID or Node Group Name	Results Output Interval of Displacement Monitoring
nout 100	node_monit_1 55	nout_monit nout_monit

Output Control Displacement	Output Control Velocity	Output Control Acceleration	Output Control Reaction Force	Output Control Strain	Output Control Stress
iout_list(1) 0	iout_list(2) 0	iout_list(3) 0	iout_list(4) 0	iout_list(5) 0	iout_list(6) 0

2-3 !BOUNDARY (Same items as in Static Analysis)

Definition of displacement boundary conditions

Node ID or Node Group Name	Start No. of Restricted Degree of Freedom	End No. of Restricted Degree of Freedom	Restricted Value
NODE_ID FIX,	DOF_idS 1,	DOF_idE 3,	Value 0.0

2-4 !CLOAD (Same items as in Static Analysis)

Definition of concentrated load

Node ID or Node Group Name	Degree of Freedom No.	Load Value
CL1,	3,	-1.0

5-4 !COUPLE, TYPE=1

Definition of coupled surface

Coupling Surface Group Name
SCOUPLE

2-11 !STEP, CONVERG=1.E-10, ITMAX=20

Control of nonlinear static analysis

(Omissible in the case of linear analysis, and unnecessary for explicit method)

Convergence Value Judgment Threshold (Default: 1.0E-06)	No. of Sub Steps (When AMP exists, AMP has priority)	Max No. of Iterative Calculations
CONVERG 1.0E-10	SUBSTEPS	ITMAX 20

### 1.7.3.7 (6) Dynamic analysis (Frequency Response Analysis) Control Data

#### 1.7.3.7.1 Example of Dynamic analysis (Frequency Response Analysis)

```

!SOLUTION, TYPE=DYNAMIC          1-2
!DYNAMIC                           5-1
  11 , 2
  14000, 16000, 20, 15000.0
  0.0 , 6.6e-5
  1, 1, 0.0 , 7.2E-7
  10, 2, 1
  1, 1, 1, 1, 1, 1
!EIGENREAD                         5-5
  eigen0.log
  1, 5
!FLOAD, LOAD CASE=2                5-6
  _PickedSet5 , 2, 1.0
!FLOAD, LOAD CASE=2
  _PickedSet6 , 2, 1.0

```

#### 1.7.3.7.2 Description of Header

- Red figures are the values indicated in the example.
- Alphabetic characters in the 2nd line of the table express the parameter name.

### 5-1 !DYNAMIC

Controlling the frequency response analysis

Solution of Equation of Motion	Analysis Types
idx_eqa	idx_resp
11	2

Minimum Frequency	Maximum Frequency	Number of divisions for the frequency range	Frequency to obtain displacement
f_start 14000	f_end 16000	n_freq 20	f_disp 15000.0

Analysis Start Time	Analysis End Time
0.0	6.6e-5

Type of Mass Matrix	Type of Damping	Parameter $R_m$ of Rayleigh Damping	Parameter $R_k$ of Rayleigh Damping
idx_mass 1	idx_dmp 1	ray_m 0.0	ray_k 7.2E-7

Results Output Interval in Time Domain	Visualization Type(1-Mode shapes,2-Time history result at f_disp)	Monitoring Node ID in Frequency Domain
nout 10	vistype 2	nodeout 1

Output ControlDisplacement	Output ControlVelocity	Output ControlAcceleration	Output Controlignored	Output Controlignored	Output Controlignored
iout_list(1) 1	iout_list(2) 1	iout_list(3) 1	iout_list(4) 1	iout_list(5) 1	iout_list(6) 1

### 5-5 !EIGENREAD

Controlling the input file for frequency response analysis

The name of eigenvalue analysis log
eigenlog_filename
eigen0.log

lowest mode to be used in frequency response analysis	highest mode to be used frequency response analysis
start_mode 1	end_mode 5

## 5-6 !FLOAD

Defining external forces applied in frequency response analysis

Node ID, Node Group Name or Surface Group Name	Degree of Freedom No.	Load Value
PickedSet5	2	1.0

### 1.7.4 Solver Control Data

#### 1.7.4.1 Example of Solver Control Data

```
### SOLVER CONTROL
!SOLVER, METHOD=CG, PRECOND=1, ITERLOG=YES, TIMELOG=YES      6-1
10000, 1                                         6-2
1.0e-8, 1.0, 0.0
```

#### 1.7.4.2 Description of Header

- Red figures are the values indicated in the example.

#### 1.7.4.2.1 6-1 !SOLVER

METHOD = method  
       (CG, BiCGSTAB, GMRES, GPBiCG, etc.)  
 TIMELOG = whether solver computation time is output  
 MPCMETHOD = method for multipoint constraints  
       (1: Penalty method,  
       2: MPC-CG method,  
       3: Explicit master-slave elimination)  
 DUMPTYPE = type of matrix dumping  
 DUMPEXIT = whether program exits right after dumping matrix

The following parameters will be disregarded when a direct solver is selected in the method.

PRECOND = preconditioner  
 ITERLOG = whether solver convergence history is output  
 SCALING = whether matrix is scaled so that each diagonal element becomes 1  
 USEJAD = whether matrix ordering optimized for vector processors is performed  
 ESTCOND = frequency for estimating condition number  
       (Estimation performed at every specified number of iterations and  
       at the last iteration. No estimation when 0 is specified.)

6-2

No. of Iterations	Iteration Count of Preconditioning	No. of Krylov Subspaces	No. of Colors for Multi-Color ordering	No. of Recycling Set-Up Info for Preconditioning
NITER 10000	iterPREMAX 1	NREST	NCOLOR_IN	RECYCLEPRE

6-3

Truncation Error	Scale Factor for Diagonal Elements when computing Preconditioning Matrix	Not Used
1.0e-8,	1.0,	0.0

## 1.7.5 Post Process (Visualization) Control Data

An example of the post process (visualization) control data and the contents are shown in the following.

### 1.7.5.1 Example of Visualization Control Data

Each description number (P1-0, P1-1, etc.) is linked to the number of the detailed descriptions in the following.

- P1- expresses the common data, and P2- expresses the parameter for the purpose of the rendering. In addition, the rendering will become valid only when the output\_type=BMP.
- When the surface\_style is !surface\_style = 2 (isosurface) !surface\_style = 3 (user specified curved surface), a separate setting is required. The data is indicated collectively after the common data. (P3- is a description of the isosurface in !surface\_style = 2. P4- is a description of the user specified curved surface in !surface\_style = 3.)
- The items indicated with two ! like “!!”, will be recognized as a comment and will not affect the analysis.

## Post Control	Description No.
!VISUAL, method=PSR	P1-0
!surface_num = 1	P1-1
!surface 1	P1-2
!surface_style = 1	P1-3
!display_method = 1	P1-4
!color_comp_name = STRESS	P1-5
!colorsubcomp_name	P1-6
!color_comp 7	P1-7
!! color_subcomp = 1	P1-8
!iso_number	P1-9
!specified_color	P1-10
!deform_display_on = 1	P1-11
!deform_comp_name	P1-12
!deform_comp	P1-13
!deform_scale = 9.9e-1	P1-14
!initial_style = 1	P1-15
!deform_style = 3	P1-16
!initial_line_color	P1-17
!deform_line_color	P1-18
!output_type = BMP	P1-19
!x_resolution = 500	P2-1
!y_resolution = 500	P2-2
!num_of_lights = 1	P2-3
!position_of_lights = -20.0, 5.8, 80.0	P2-4
!viewpoint = -20.0 -10.0 5.0	P2-5
!look_at_point	P2-6
!up_direction = 0.0 0.0 1.0	P2-7
!ambient_coef= 0.3	P2-8
!diffuse_coef= 0.7	P2-9
!specular_coef= 0.5	P2-10
!color_mapping_style= 1	P2-11
!! interval_mapping_num	P2-12
!interval_mapping= -0.01, 0.02	P2-13
!rotate_style = 2	P2-14
!rotate_num_of_frames	P2-15
!color_mapping_bar_on = 1	P2-16
!scale_marking_on = 1	P2-17
!num_of_scale = 5	P2-18
!font_size = 1.5	P2-19
!font_color = 1.0 1.0 1.0	P2-20
!background_color	P2-21
!isoline_color	P2-22
!boundary_line_on	P2-23
!color_system_type	P2-24
!fixed_range_on = 1	P2-25

!range\_value = -1.E-2, 1.E-2

P2-26

#### 1.7.5.1.1 Common Data List<P1-1 - P1-19>

No.	Keywords	Types	Contents
P1-0	!VISUAL		Specification of the visualization method
P1-1	surface_num		No. of surfaces in one surface rendering
P1-2	surface		Setting of the contents of surface
P1-3	surface_style	integer	Specification of the surface type (Default: 1): Boundary surface2: Isosurface3: Curved surface defined by user based on the equation
P1-4	display_method	integer	Display method (Default: 1). Color code display2. Boundary line display3. Color code and boundary line display4. Display of 1 specified color5. Isopleth line display by classification of color
P1-5	color_comp_name	character(100)	Compatible with parameter name and colormap(Default: 1st parameter name)
P1-6	color_subcomp_name	character(4)	When the parameter is a vector, specifies the component to be displayed. (Default: x)norm: Norm of the vectorx: x componenty: y componentz: z component
P1-7	color_comp	integer	Provides an ID number to the parameter name(Default: 0)

No.	Keywords	Types	Contents
P1-8	color_subcomp	integer	When the degree of freedom of the parameter is 1 or more, specifies the degree of freedom number to be displayed.0: Norm(Default: 1)
P1-9	iso_number	integer	Specifies the number of isopleth lines.(Default:5)
P1-10	specified_color	real	Specified the color when the display_method = 40.0 < specified_color < 1.0
P1-11	!deform_display_on	integer	Specifies the existence of deformation.1: On, 0: Off (Default: 0)
P1-12	!deform_comp_name	character(100)	Specifies the attribution to be adopted when specifying deformation.(Default: Parameter called DISPLCEMENT)
P1-13	!deform_comp	integer	ID number of the parameter when specifying deformation.(Default: 0)
P1-14	!deform_scale	real	Specifies the displacement scale when displaying deformation.Default:Autostandard_scale = 0.1 * sqrt(x_range2 + y_range2 + z_range2) / max_deformuser_defined: real_scale = standard_scale * deform_scale

No.	Keywords	Types	Contents
P1-15	! initial_style	integer	Specifies the type of deformation display.(Default: 1)0: Not specified1: Solid line mesh(Displayed in blue if not specified)2: Gray filled pattern3: Shading (Let the physical attributions respond to the color)4: Dotted line mesh (Displayed in blue if not specified)
P1-16	!deform_style	integer	Specifies the shape display style after the initial deformation.(Default: 4)0: Not specified1: Solid line mesh (Displayed in blue if not specified)2: Gray filled pattern3: Shading (Let the physical attributions respond to the color)4: Dotted line mesh (Displayed in blue if not specified)
P1-17	! initial_line_color	real (3)	Specifies the color when displaying the initial mesh. This includes both the solid lines and dotted lines.(Default: Blue(0.0, 0.0, 1.0))
P1-18	!deform_line_color	real (3)	Specifies the color when displaying the deformed mesh. This includes both the solid lines and dotted lines.(Yellow(1.0, 1.0, 0.0))

No.	Keywords	Types	Contents
P1-19	output_type	character(3)	Specifies the type of output file. (Default: AVS)AVS: UCD Data for AVS(only on object surface)BMP: Image data (BMP format)COMPLETE_AVSS: UCD data for AVSCOMPLETE_REORDER Rearranges the node and element IDSEPARATE_COMPLETE For each decomposed domainCOMPLETE_MICROAVS Outputs the physical value scalarFSTR_FEMAP_NEUTR Neutral file for FEMAP

#### 1.7.5.1.2 Rendering Data List <P2-1 - P2-26>

(Valid only when the output\_type=BMP)

	Keywords	Types	Contents
P2-1	x_resolution	integer	Specifies the width of final figure. (Default: 512)
P2-2	y_resolution	integer	Specifies the height of final figure. (Default: 512)
P2-3	num_of_lights	integer	Specifies the number of lights. (Default: 1)
P2-4	position_of_lights	real(:)	Specifies the position of the lights by coordinates.(Default: Directly above front)Specification method !position_of_lights= x, y, z, x !position_of_lights=100.0, 200.
P2-5	viewpoint	real(3)	Specifies the viewpoint position by coordinates.(Default: x = (xmin + xmax)/2.0) y = ymin + 1.5 *( ymax - ymin)z = zmin + 1.5 *( zmax - zmin )

	Keywords	Types	Contents
P2-6	look_at_point	real(3)	Specifies the look at point position.(Default: Center of data)
P2-7	up_direction	real(3)	Defines the view frame at Viewpoint, look_at_point and up_direction(Default: 0.0, 0.0, 1.0)
P2-8	ambient_coef	real	Specifies the peripheral brightness. (Default: 0.3)
P2-9	diffuse_coef	real	Specifies the intensity of the diffused reflection light by coefficient.(Default: 0.7)
P2-10	specular_coef	real	Specifies the intensity of specular reflection by coefficient.(Default: 0.6)
P2-11	color_mapping_style	integer	Specifies the color mapping style.(Default: 1)1: Complete linear mapping (Maps overall color in RGB linear)2: Clip linear mapping (Maps from mincolor to maxcolor in the RGB color space)3: Nonlinear color mapping (Partitions all domains into multiple sections, and performs linear mapping for each section)4: Optimum auto adjustment (Performs a statistical process of the data distribution to determine the color mapping)
P2-12	interval_mapping_num	integer	Specifies the number of sections when the color_mapping_style = 3

	Keywords	Types	Contents
P2-13	interval_mapping	real(:)	Specifies the section position and color number when the color_mapping_style = 2 or 3.if the color_mapping_style=2;!interval the color_mapping_style=3;!interval Must be describe in one line.
P2-14	rotate_style	integer	Specifies the rotating axis of animation.1: Rotates at x-axis.2: Rotates at y-axis.3: Rotates at z axis.4: Particularly, specifies the viewpoint to perform animation. (8 frames)
P2-15	rotate_num_of_frames	integer	Specifies the cycle of animation.(rotate_style = 1, 2, 3)(Default: 8)
P2-16	color_mapping_bar_on	integer	Specifies the existence of color mapping bar.0: off; 1: on; Default: 0
P2-17	scale_marking_on	integer	Specifies whether to display the value on the color mapping bar.0: off; 1: on; Default: 0
P2-18	num_of_scale	integer	Specifies the number of memories of the color bar. (Default: 3)
P2-19	font_size	real	Specifies the font size when displaying the value of the color mapping bar. Range: 1.0-4.0 (Default: 1.0)
P2-20	font_color	real(3)	Specifies the display color when displaying the value of the color mapping bar. (Default: 1.0, 1.0, 1.0 (White))

	Keywords	Types	Contents
P2-21	background_color	real(3)	Specifies the background color. (Default: 0.0, 0.0, 0.0 (Black))
P2-22	isoline_color	real(3)	Specifies the color of the isopleth line. (Default: Same color as the value)
P2-23	boundary_line_on	integer	Specifies whether to display the zone of the data.0: off; 1: on; Default: 0
P2-24	color_system_type	integer	Specifies the color mapping style.(Default: 1)1: (Blue - Red)(in ascending order)2: Rainbow mapping (Ascending order from red to purple)3: (Black - White)(in ascending order)
P2-25	fixed_range_on	integer	Specifies whether to maintain the color mapping style for other time steps.0: off; 1: on; (Default: 0)
P2-26	range_value	real(2)	Specifies the section.

#### 1.7.5.1.3 Data List by Setting Values of `surface_style`

In the case of isosurface (`surface_style=2`)

	Keywords	Types	Contents
P3-1	data_comp_name	character(100)	Provides the name to the attribution of the isosurface.
P3-2	data_subcomp_name	character(4)	When the parameter is a vector, specifies the component to be displayed. (Default: x)norm: Norm of the vectorx: x componenty: y componentz: z component
P3-3	data_comp	integer	Provides an ID number to the parameter name(Default: 0)
P3-4	data_subcomp	integer	When the degree of freedom of the parameter is 1 or more, specifies the degree of freedom number to be displayed.0: Norm(Default: 1)
P3-5	iso_value	real	Specifies the value of the isosurface.

In the case of a curved surface (`surface_syle = 3`) specified by the equation of the user

	Keywords	Types	Contents
P4-1	method	integer	Specifies the attribution of the curved surface. (Default: 5)1: Spherical surface2: Ellipse curved surface3: Hyperboloid4: Paraboloid5: General quadric surface
P4-2	point	real(3)	Specifies the coordinates of the center when method = 1, 2, 3, or 4(Default: 0.0, 0.0, 0.0)

	Keywords	Types	Contents
P4-3	radius	real	Specifies the radius when method = 1(Default: 1.0)
P4-4	length	real	Specifies the length of the diameter when method = 2, 3, or 4. Note: The length of one diameter in the case the ellipse curved surface is 1.0.
P4-5	coef	real	Specifies the coefficient of a quadric surface when method=5.coef[1]x <sup>2</sup> + coef[2]y <sup>2</sup> + coef[3]z <sup>2</sup> + coef[4]xy + coef[5]xz + coef[6]yz + coef[7]x + coef[8]y + coef[9]z + coef[10]=0Ex: coef=0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0, -10.0This means the plain surface of y=10.0

## 1.7.6 Details of Analysis Control Data Parameters

The details of each parameter explained in item 7.3 are described in the following.

The analysis control data is classified as follow.

1. Common control data
2. Control data for static analysis
3. Control data for eigenvalue analysis
4. Control data for heat conduction analysis
5. Control data for dynamic analysis
6. Solver control data
7. Post process (visualization) control data

### 1.7.6.1 Common Control Data

#### 1.7.6.1.1 (1) !VERSION (1-1)

Specifies the solver version number. The current version number is 5.

Example of Use

```
!VERSION
5
```

#### 1.7.6.1.2 (2) !SOLUTION (1-2)

Specifies the type of analysis.

Parameter

```
TYPE=
STATIC      : Linear static analysis
NLSTATIC    : Nonlinear static analysis
HEAT        : Heat conduction analysis
EIGEN       : Eigenvalue analysis
DYNAMIC     : Dynamic analysis
STATICEIGEN : Nonlinear static analysis & rarr; Eigenvalue analysis
ELEMCHECK   : Element shape check
```

Example of Use

```
!SOLUTION, TYPE=STATIC
```

#### 1.7.6.1.3 (3) !WRITE, VISUAL (1-3)

Specifies the output data by the visualizer via memory.

Parameter

```
FREQUENCY = step interval of output (Default: 1)
```

Example of Use

```
!WRITE, VISUAL, FREQUENCY=2
```

#### 1.7.6.1.4 (4) !WRITE, RESULT (1-4)

Specifies the output of the analysis results file.

Parameter

```
FREQUENCY = step interval of output (Default:1)
```

Example of Use

```
!WRITE, RESULT, FREQUENCY=2
```

#### 1.7.6.1.5 (5) !WRITE, LOG (1-5)

Specifies the step interval for output to the log file.

Parameter

```
FREQUENCY = step interval of output (Default:1)
```

Example of Use

```
!WRITE, LOG, FREQUENCY=2
```

#### 1.7.6.1.6 (6) !OUTPUT\_VIS (1-6)

Output item control of the visualization

```
!WRITE, VISUAL must be specified
```

Parameter

N/A

2nd Line or later

```
(2nd line or later) Parameter name, ON/OFF
```

The following parameter names can be specified.

Parameter Names	Physical Values
DISP	Displacement (Default output)
ROT	Rotation (Only for 781,761 shell)
REACTION	Reaction force of nodes
NSTRAIN	Strain of nodes
NSTRESS	Stress of nodes (Default output)
NMISES	Mises stress of nodes (Default output)
TH_NSTRAIN	Thermal strain of nodes (Not included)
VEL	Velocity
ACC	Acceleration

Example of Use

```
!OUTPUT_VIS  
NSTRAIN, ON  
NSTRESS, OFF
```

#### 1.7.6.1.7 (7) !OUTPUT\_RES (1-7)

Output item control of the result

!WRITE, RESULT must be specified

Parameter

N/A

2nd Line or later

(2nd line or later) Parameter name, ON/OFF

The following parameter names can be specified.

Parameter Names	Physical Values
DISP	Displacement (Default output)
ROT	Rotation (Only for 781,761 shell)
REACTION	Reaction force of nodes
NSTRAIN	Strain of nodes
NSTRESS	Stress of nodes (Default output)
NMISES	Mises stress of nodes (Default output)
ESTRAIN	Strain of elements
ESTRESS	Stress of elements (Default output)
EMISES	Mises stress of elements (Default output)
ISTRAIN	Strain of integration points
ISTRESS	Stress of integration points
PL_ISTRAIN	Plastic strain of integration points
TH_NSTRAIN	Thermal strain of nodes (Not included)
TH_ESTRAIN	Thermal strain of elements (Not included)
TH_ISTRAIN	Thermal strain of integration points (Not included)
VEL	Velocity
ACC	Acceleration
TEMP	Temperature
PRINC_NSTRESS	Nodal principal stress(Scalar value)
PRINCV_NSTRESS	Nodal principal stress(Vector value)
PRINC_NSTRAIN	Nodal principal strain(Scalar value)
PRINCV_NSTRAIN	Nodal principal strain(Vector value)
PRINC_ESTRESS	Elemental principal stress(Scalar value)
PRINCV_ESTRESS	Elemental principal stress(Vector value)
PRINC_ESTRAIN	Elemental principal strain(Scalar value)
PRINCV_ESTRAIN	Elemental principal strain(Vector value)
SHELL_LAYER	Output per layer of layered shell element
SHELL_SURFACE	Output of surface information of shell element

Example of Use

```
! OUTPUT_RES  
ESTRESS, OFF  
ISTRESS, ON
```

#### 1.7.6.1.8 (8) !RESTART (1-8)

Controls the writing of the restart file. When not specified, the restart file can not be written.

Parameter

```
FREQUENCY = n      : step interval of output (Default: 0)  
                  n > 0    : Output for each n step  
                  n < 0    : First, reads the restart file, then outputs for each n step
```

Example of Use

!RESTART, FREQUENCY==2

#### 1.7.6.1.9 (9) !ECHO (1-9)

Outputs the node data, element data and material data to the log file.

Parameter

N/A

#### 1.7.6.1.10 (10) !ORIENTATION (1-10)

Definition of local coordinate system

Parameter

NAME = Name of local coordinate system

DEFINITION = COORDINATES (Default)/NODES

#### 1.7.6.1.11 2nd Line of later

- In case of DEFINITION=COORDINATES

(2nd line or later) a1, a2, a3, b1, b2, b3, c1, c2, c3

- In case of DEFINITION=NODES

(2nd line or later) a, b, c

Parameter Name	Attributions	Contents
a1, a2, a3	R	coordinate of point a
b1, b2, b3	R	coordinate of point b
c1, c2, c3	R	coordinate of point c
a,b,c	I	Node ID of a,b,c, respectively

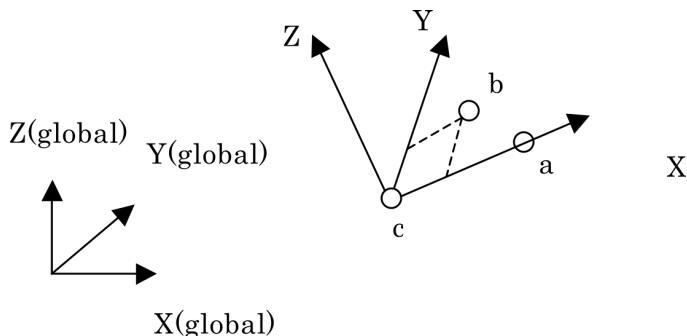


Figure 18: Analysis Control Data

#### 1.7.6.1.12 (11) !SECTION (1-11)

Definition of local coordinate system the section correspondent to.

Parameter

SECTNUM = Index of section defined in M1–10 in chapter 6.3.

ORIENTATION = Name of local coordinate system defined in (1–10) above.

2nd Line or later

N/A

### **1.7.6.1.13 (12) !INITIAL\_CONDITION (1-12)**

Definition of initial condition

Parameter

TYPE = TEMPERATURE/VELOCITY/ACCELERATION

In case of TYPE = TEMPERATURE

(2nd line) ng1, t1

(3rd line or later) ng2, t2

...

Parameter Name	Attributions	Contents
ng1,ng2, ...	C/I	name of node group/index of node
t1, t2, ...	R	temperature

In case of TYPE= VELOCITY/ACCELERATION

(2nd line) ng1, dof1, v1

(3rd line or later) ng2, dof2, v2

...

Parameter Name	Attributions	Contents
ng1,ng2, ...	C/I	name of node group/index of node
dof1, dof2, ...	I	dof number(1-6)
v1, v2, ...	R	velocity/acceleration

### **1.7.6.1.14 (13) !END (1-13)**

Displays the end of the control data.

Parameter

N/A

## **1.7.6.2 Control Data for Static Analysis**

### **1.7.6.2.1 (1) !STATIC (2-1)**

Performs the static analysis. (Default: ommissible)

Parameter

N/A

### **1.7.6.2.2 (2) !MATERIAL (2-2)**

Definition of material physical properties

The definition of the material physical properties is used in a set with the !MATERIAL and the !ELASTICITY, !PLASTICITY and etc. entered next. The !ELASTICITY, !PLASTICITY and etc. entered before !MATERIAL will be disregarded.

Note: When the !MATERIAL is defined in the analysis control data, the !MATERIAL definition in the mesh data will be disregarded. When the !MATERIAL is not defined in the analysis control data, the !MATERIAL definition in the mesh data is used.

Parameter

NAME = Material name

### 1.7.6.2.3 (3) !ELASTIC (2-2-1)

Definition of elastic material

Parameter

TYPE = ISOTROPIC (Default)/ ORTHOTROPIC / USER  
DEPENDENCIES = 0 (Default)/1

2nd Line or later

- In the case of TYPE = ISOTROPIC

(2nd Line) YOUNGS, POISSON, Temperature

Parameter Name	Attributions	Contents
YOUNGS	R	Young's Modulus
POISSON	R	Poisson's Ratio
Temperature	R	Temperature (required when DEPENDENCIES = 1)

- In case of TYPE=ORTHOTROPIC

(2nd Line) E1, E2, E3, 12, 13, 23, G12, G13, G23, Temperature

$$\begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{23} \\ 2\varepsilon_{31} \end{bmatrix} = \begin{bmatrix} 1/E_1 & -\nu_{12}/E_1 & -\nu_{13}/E_1 & 0 & 0 & 0 \\ & 1/E_2 & -\nu_{23}/E_2 & 0 & 0 & 0 \\ & & 1/E_3 & 0 & 0 & 0 \\ & & & 1/G_{12} & 0 & 0 \\ & & & & 1/G_{23} & 0 \\ & & & & & 1/G_{31} \end{bmatrix} \begin{bmatrix} \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{12} \\ \sigma_{23} \\ \sigma_{31} \end{bmatrix} \quad (1)$$

- In the case of TYPE=USER

(2nd line – 10th line) v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

### 1.7.6.2.4 (4) !PLASTIC (2-2-2)

Definition of plastic material

Parameter

YIELD = MISES (Default), Mohr–Coulomb, DRUCKER-PRAGER, USER  
HARDEN = BILINEAR (Default), MULTILINEAR, SWIFT, RAMBERG-OSGOOD,  
KINEMATIC, COMBINED  
DEPENDENCIES = 0 (Default)/1

\*\* 2nd line or later \*\*

In case of YIELD = MISES (Default)

\*\* In case of HARDEN = BILINEAR (Default) \*\*

(2nd line) YIELD0, H

\*\* In case of HARDEN = MULTILINEAR \*\*

(2nd line) YIELD, PSTRAIN, Temperature  
(3rd line) YIELD, PSTRAIN, Temperature

... continues

\*\* In case of HARDEN = SWIFT \*\*  
(2nd line) \$\\epsilon\_0 , K, n

\*\* In case of HARDEN = RAMBERG–OSGOOD \*\*  
(2nd line) \$\\epsilon\_0 , D, n

\*\* In case of HARDEN = KINEMATIC \*\*  
(2nd line) YIELD0, C

\*\* In case of HARDEN = COMBINED \*\*  
(2nd line) YIELD0, H, C

In case of YIELD = Mohr-Coulomb or Drucker-Prager  
\*\* In case of HARDEN = BILINEAR(Default) \*\*  
(2nd line) c, FAI, H

\*\* In case of HARDEN = MULTILINEAR \*\*  
(2nd line) FAI  
(3rd line) PSTRAIN, c  
(4th line) PSTRAIN, c  
... continues

HARDEN =others will be disregarded, becomes the default (BILINEAR).

Parameter Name	Attribu-tions	Contents
YIELD0	R	Initial yield stress
H	R	Hardening factor
PSTRAIN	R	Plastic strain
YIELD	R	Yield stress
$\epsilon_0, K, n$	R	$\bar{\sigma} = k (\epsilon_0 + \bar{\epsilon})^n$
$\epsilon_0, D, n$	R	$\epsilon = \frac{\sigma}{E} + \epsilon_0 \left(\frac{\sigma}{D}\right)^n$
FAI	R	Internal frictional angle
c	R	Viscosity
C	R	Linear motion hardening factor
Tempearture	R	Temperature (required when DEPENDENCIES = 1)
v1, v2...v10	R	Material constant

\*\* In the case of YIELD= USER \*\*  
(2nd Line or later) v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

Example of Use

```
!PLASTIC, YIELD=MISES, HARDEN=MULTILINEAR, DEPENDENCIES=1
276.0, 0.0, 20.
296.0, 0.0018, 20.
299.0, 0.0053, 20.
303.0, 0.008, 20.
338.0, 0.0173, 20.
372.0, 0.0271, 20.
400.0, 0.037, 20.
419.0, 0.0471, 20.
437.0, 0.0571, 20.
450.0, 0.0669, 20.
```

```

460.0, 0.0767, 20.
469.0, 0.0867, 20.
477.0, 0.0967, 20.
276.0, 0.0, 100.
276.0, 0.0018, 100.
282.0, 0.0053, 100.
295.0, 0.008, 100.
330.0, 0.0173, 100.
370.0, 0.0271, 100.
392.0, 0.037, 100.
410.0, 0.0471, 100.
425.0, 0.0571, 100.
445.0, 0.0669, 100.
450.0, 0.0767, 100.
460.0, 0.0867, 100.
471.0, 0.0967, 100.
128.0, 0.0, 400.
208.0, 0.0018, 400.
243.0, 0.0053, 400.
259.0, 0.008, 400.
309.0, 0.0173, 400.
340.0, 0.0271, 400.
366.0, 0.037, 400.
382.0, 0.0471, 400.
396.0, 0.0571, 400.
409.0, 0.0669, 400.
417.0, 0.0767, 400.
423.0, 0.0867, 400.
429.0, 0.0967, 400.

```

The work hardening coefficient will be calculated by inserting the data from the above inputdata, regarding the specified temperature or plastic strain. It is necessary to input the same PSTRAIN array for each temperature.

#### **1.7.6.2.5 (5) !HYPERELASTIC (2-2-3)**

Definition of hyperelastic material

Parameter

TYPE = NEOHOOKE ( Default )  
 MOONEY-RIVLIN  
 ARRUDA-BOYCE  
 USER

2nd Line or later

In the case of TYPE = NEOHOOKE

(2nd line) C10, D

Parameter Name	Attributions	Contents
C10	R	Material constant
D	R	Material constant

In case of TYPE = MOONEY-RIVLIN

(2nd line) C10, C01, D

Parameter Name	Attributions	Contents
C10	R	Material constant
C01	R	Material constant
D	R	Material constant

In case of TYPE = ARRUDA-BOYCE

(2nd line) mu, lambda\_m, D

Parameter Name	Attributions	Contents
mu	R	Material constant
lambda_m	R	Material constant
D	R	Material constant

In case of TYPE = USER

(2nd line - 10th line) v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

#### 1.7.6.2.6 (6) !VISCOELASTIC (2-2-4)

Definition of viscoelastic material

Parameter

DEPENDENCIES = the number of parameters depended upon (Not included)

\*\* 2nd Line or later \*\*

(2nd line) g, t

Parameter Name	Attributions	Contents
g	R	Shear relaxation modulus
t	R	Relaxation time

#### 1.7.6.2.7 (7) !CREEP (2-2-5)

Definition of creep material

Parameter

TYPE = NORTON (Default)

DEPENDENCIES = 0 (Default) / 1

\*\* 2nd Line or later \*\*

(2nd line) A, n, m, Tempearature

Parameter Name	Attributions	Contents
A	R	Material modulus
n	R	Material modulus
m	R	Material modulus
Tempearture	R	Temperature(required when DEPENDENCIES=1)

#### 1.7.6.2.8 (8) !DENSITY (2-2-6)

Definition of mass density

Parameter

DEPENDENCIES = the number of parameters depended upon (Not included)

\*\* 2nd Line or later \*\*

(2nd line) density

Parameter Name	Attributions	Contents
density	R	Mass density

#### 1.7.6.2.9 (9) !EXPANSION\_COEFF (2-2-7)

Definition of coefficient of linear expansion

The coefficient to be input here is not the coefficient of linear expansion  $\alpha$  at each temperature, but its averaged value between the reference temperature  $T_{ref}$  and each temperature  $T$  as follows:

$$\bar{\alpha}(T) = \frac{1}{T - T_{ref}} \int_{T_{ref}}^T \alpha(T) dT \quad (2)$$

Parameter

TYPE = ISOTROPIC( Default ) / ORTHOTROPIC

DEPENDENCIES = 0( Default ) / 1

\*\* 2nd Line or later \*\*

In case of TYPE=ISOTROPIC

(2nd line) expansion , Temperature

In case of TYPE=ORTHOTROPIC

(2nd line) \$\alpha\$11 , \$\alpha\$22 , \$\alpha\$33 , Temperature

Parameter Name	Attributions	Contents
expansion	R	Coefficient of thermo expansion
\$ \$11, \$ \$22, \$ \$33	R	Coefficient of thermo expansion
Tempearture	R	Temperature (required when DEPENDENCIES = 1)

#### 1.7.6.2.10 (10) !TRS (2-2-8)

Thermorheological Simplicity description on temperature behavior of viscoelastic materials

Parameter

DEFINITION = WLF( Default ) /ARRHENUS

\*\* 2nd line or later \*\*

(2nd line) \$ \_0\$, C1, C2

Parameter Name	Attributions	Contents
\$ _0\$	R	Reference temperature
C1, C2	R	Material constants

### **1.7.6.2.11 (11) !FLUID (2-2-9)**

Definition of flow condition

Parameter

TYPE = INCOMP\_NEWTONIAN ( Default )

\*\* 2nd Line or later \*\*

(2nd line) mu

Parameter Name	Attributions	Contents
mu	R	Viscosity

### **1.7.6.2.12 (12) !USER/MATERIAL (2-2-10)**

Input of user defined material

Parameter

NSTATUS = Specifies the number of state variables of material (Default: 1)

\*\* 2nd line or later \*\*

(2nd line-10th line) v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

### **1.7.6.2.13 (13) !BOUNDARY (2-3)**

Definition of displacement boundary conditions

Parameter

GRPID = Group ID

AMP = Time function name (Specified in !AMPLITUDE, valid in dynamic analysis)

ROT\_CENTER = Node number of rotational constraint or node group name.

When specified it, this '!BOUNDARY' is recognized as rotational constraint.

\*\* 2nd line or later \*\*

(2nd line) NODE\_ID, DOF\_idS, DOF\_idE, Value

Parameter Name	Attributions	Contents
NODE_ID	I/C	Node ID or node group name
DOF_idS	I	Start No. of restricted degree of freedom
DOF_idE	I	End No. of restricted degree of freedom
Value	R	Restricted value (Default: 0)

Example of Use

!BOUNDARY, GRPID=1

1, 1, 3, 0.0

ALL, 3, 3,

Note: Restricted value is 0.0

### **1.7.6.2.14 (14) !SPRING (2-3-1)**

Definition of spring boundary conditions

Parameter

GRPID = Group ID

\*\* 2nd line or later \*\*

(2nd line) NODE\_ID, DOF\_id, Value

Parameter Name	Attributions	Contents
NODE_ID	I/C	Node ID or node group name
DOF_id	I	Restricted degree of freedom
Value	R	Spring constant

Example of Use

!SPRING, GRPID=1

1, 1, 0.5

### 1.7.6.2.15 (15) !CLOAD (2-4)

Definition of concentrated load

Parameter

GRPID = Group ID

AMP = Time function name (Specified in !AMPLITUDE, valid in dynamic analysis)

ROT\_CENTER = Node number of rotational constraint or node group name.

When specified it, this `!CLOAD` is recognized as load of torque.

\*\* 2nd line or later \*\*

(2nd line) NODE\_ID, DOF\_id, Value

Parameters	Attributions	Contents
NODE_ID	I/C	Node ID or node group name
DOF_id	I	Degree of freedom No.
Value	R	Load value

Example of Use

!CLOAD, GRPID=1

1, 1, 1.0e3

ALL, 3, 10.0

!CLOAD, ROT\_CENTER=7, GRPID=1

TORQUE\_NODES, 1, 3

TORQUE\_NODES, 3, -4

### 1.7.6.2.16 (16) !DLOAD (2-5)

Definition of distributed load

Parameter

GRPID = Group ID

AMP = Time Function Name (Specified in !AMPLITUDE, valid in dynamic analysis)

FOLLOW = YES(Default) / NO

(whether pressure load follow deformation, valid in finite displacement analysis)

\*\* 2nd Line or later \*\*

(2nd line) ID\_NAME, LOAD\_type, param1, param2, ...

Parameter Name	Attributions	Contents
ID_NAME	I/C	Surface group name, element group name, or element ID
LOAD_type	C	Load type No.
param*	R	Load parameter (refer to following)

## Load Parameters

Load Type No.	Types	No. of Parameters	Parameter Array & Meaning
S	Applies pressure to surface specified in the surface group	1	Pressure value
P0	Applies pressure to shell element	1	Pressure value
PX	Pressure to shell element along X direction	1	Pressure value
PY	Pressure to shell element along Y direction	1	Pressure value
PZ	Pressure to shell element along Z direction	1	Pressure value
P1	Applies pressure to 1st surface	1	Pressure value
P2	Applies pressure to 2nd surface	1	Pressure value
P3	Applies pressure to 3rd surface	1	Pressure value
P4	Applies pressure to 4th surface	1	Pressure value
P5	Applies pressure to 5th surface	1	Pressure value
P6	Applies pressure to 6th surface	1	Pressure value
BX	Body force in X direction	1	Body force value
BY	Body force in Y direction	1	Body force value
BZ	Body force in Z direction	1	Body force value
GRAV	Gravity	4	Gravitaional acceleration, gravity direction cosine
CENT	Centrifugal force	7	Angular velocity, position vector at a point on the rotation axis, vector in the rotating axis direction

## Example of Use

```
!DLOAD, GRPID=1
  1, P1, 1.0
  ALL, BX, 1.0
  ALL, GRAV, 9.8, 0.0, 0.0, -1.0
  ALL, CENT, 188.495, 0.0, 0.0, 0.0, 0.0, 1.0
```

### 1.7.6.2.17 (17) !ULOAD (2-6)

Input of user definition load

Parameter

FILE = file name (Mandatory)

### 1.7.6.2.18 (18) !CONTACT\_ALGO (2-7)

Specification of the contact analysis algorithm

Parameter

TYPE = SLAGRANGE (Lagrange multiplier method)  
ALAGRANGE (Extended Lagrange multiplier method)

### 1.7.6.2.19 (19) !CONTACT (2-8)

Definition of contact conditions

Parameter

GRPID = Boundary conditions group ID  
INTERACTION = SSLID(Default) / FSLID  
NTOL = Contact normal direction convergence threshold (Default: 1.e-5)  
TTOL = Contact tangential direction convergence threshold (Default: 1.e-3)  
NPENALTY = Contact normal direction Penalty (Default: stiffness matrix 1.e3)  
TPENALTY = Contact tangential direction Penalty (Default: 1.e3)

2nd line or later

(2nd line) PAIR\_NAME, fcoef , factor

Parameter Name	Attributions	Contacts
PAIR_NAME	C	Contact pair name (Defined in !CONTACT_PAIR)
fcoef	R	Friction coefficient (Default: 0.0)
factor	R	Friction penalty stiffness

Example of Use

```
!CONTACT_ALGO, TYPE=SLAGRANGE
!CONTACT, GRPID=1, INTERACTION=FSLID
    CP1, 0.1, 1.0e+5
```

### 1.7.6.2.20 (20) !TEMPERATURE (2-9)

Specification of nodal temperature used for thermal stress analysis

Parameter

READRESULT = Number of result steps of heat conduction analysis.  
When specified, the temperature is sequentially input from  
the results file of the heat conduction analysis,  
and the 2nd line and later will be disregarded.  
SSTEP = First step number that performs the reading  
of the heat conduction analysis results (Default: 1)  
INTERVAL = Step interval that performs the reading  
of the heat conduction analysis results (Default: 1)  
READTYPE = STEP(Default) / TIME  
When TIME is specified, analysis time of the stress  
analysis is synchronized with the heat conduction  
analysis (value of INTERVAL is ignored, and the  
temperature is linearly interpolated from results of the  
heat conduction analysis right before and after the  
current analysis time)

When unsteady heat conduction analysis using auto time increment was performed, and the results were output at specified time points using !TIME\_POINTS, READTYPE=TIME needs to be specified because the step interval of the results is not constant.

\*\* 2nd line or later \*\*

(2nd line) NODE\_ID, Temp\_Value

Parameter Name	Attributions	Contents
NODE_ID	I/C	Node ID or node group name
Temp_Value	R	Temperature (Default: 0)

Example of Use

```
!TEMPERATURE
  1, 10.0
  2, 120.0
  3, 330.0
!TEMPERATURE
  ALL, 20.0
!TEMPERATURE, READRESULT=1, SSTEP=1
```

#### 1.7.6.2.21 (21) !REFTEMP (2-10)

Definition of reference temperature in thermal stress analysis

Parameter

N/A

\*\* 2nd line or later \*\*

(2nd line) Value

Parameter Name	Attributions	Contents
Value	R	Reference temperature (Default: 0)

#### 1.7.6.2.22 (22) !STEP (2-11)

Setting of analysis steps

Setting is mandatory in the nonlinear static analysis and nonlinear dynamic analysis.

When this definition is omitted in analyses other than the above, all the boundary conditions will become valid and is calculated in 1 step.

When the material characteristics have viscoelasticity and creep, specify TYPE=VISCO and set the computation time conditions.

Parameter

```
TYPE      = STATIC ( default ) / VISCO
            ( semi - static analysis )
SUBSTEPS = Number of substeps of the boundary conditions
            ( Default: 1 )
CONVERG   = Convergence judgment threshold
            ( Default: 1.0e-6 )
MAXITER  = Maximum number of iterative calculations in nonlinear analysis
            ( Default: 50 )
AMP       = Time function name
            ( specified in !AMPLITUDE )
```

\*\* 2nd line or later \*\*

(2nd line) DTIME, ETIME (specified when TYPE=VISCO)

Parameter Name	Attribution	Contents
DTIME	R	Time increment value (Default: 1)
ETIME	R	End value of time increment in this step (Default: 1)

\*\* 3rd line or later \*\*

BOUNDARY, id	GRPID defined in id=!BOUNDARY
LOAD, id	GRPID defined in id!=!CLOAD, !DLOAD, !SPRING, !TEMPERATURE
CONTACT, id	GRPID defined in id!=!CONTACT

Parameter

```
!STEP, CONVERG=1.E-8
  0.1, 1.0
BOUNDARY, 1
LOAD, 1
CONTACT, 1
```

### 1.7.6.3 Control Data for Eigenvalue Analysis

#### 1.7.6.3.1 (1) !EIGEN (3-1)

Parameter settings of eigenvalue analysis

Parameter

N/A

\*\* 2nd line or later \*\*

```
(2nd line) NGET, LCZTOL, LCZMAX
```

Parameter Name	Attributions	Contents
NSET	I	No. of eigenvalue
LCZTOL	R	Allowance (Default: 1.0e-8)
LCZMAX	I	Max No. of iterations (Default: 60)

Example of Use

```
!EIGEN
  3, 1.0e-10, 40
```

### 1.7.6.4 Control Data for Heat Conduction Analysis

#### 1.7.6.4.1 (1) !HEAT (4-1)

Definition of control data regarding calculation

Parameter

TIMEPOINTS = Time list name (specify with !TIME\_POINTS, NAME)

\*\* 2nd line or later \*\*

```
(2nd line) DT, ETIME, DTMIN, DELTMX, ITMAX, ESP
```

Parameter Name	Attributions	Contents
DT	R	Initial time increment 0: Steady calculation > 0: Unsteady calculation
ETIME	R	Unsteady calculation time (mandatory for unsteady calculation)
DTMIN	R	Minimum time increment 0: Fixed time increment > 0: Auto time increment
DELTMX	R	Allowable change in temperature
ITMAX	I	Maximum number of iterations of nonlinear calculation (Default: 20)
EPS	R	Convergence judgment value (Default: 1.0e-6)

Example of Use

```
!HEAT
  (No data)      ——— Steady calculation
!HEAT
  0.0           ——— Steady calculation
!HEAT
  10.0 , 3600.0   ——— Fixed time increment unsteady calculation
!HEAT
  10.0 , 3600.0 , 1.0   ——— Auto time increment unsteady calculation
!HEAT
  10.0 , 3600.0 , 1.0 , 20.0 ——— Auto time increment unsteady calculation
```

Remarks

Only when performing auto time increment unsteady calculation, TIMEPOINTS parameter can be used to specify time points at which results and/or visualization files are output.

#### 1.7.6.4.2 (2) !FIXTEMP (4-2)

Definition of fixed temperature

Parameter

AMP = Flux history table name (specified in !AMPLITUDE)

\*\* 2nd line or later \*\*

(2nd line) NODE\_GRP\_NAME, Value

Parameter Name	Attributions	Contents
NODE_GRP_NAME	C/I	Node group name or node ID
Value	R	Temperature (Default: 0)

Example of Use

```
!FIXTEMP
  ALL, 20.0
!FIXTEMP, AMP=FTEMP
  ALL, 1.0
```

#### 1.7.6.4.3 (3) !CFLUX (4-3)

Definition of centralized heat flux given to the node

Parameter

AMP = Flux history table name (specified in !AMPLITUDE)

\*\* 2nd line or later \*\*

(2nd line) NODE\_GRP\_NAME, Value

Parameter Name	Attributions	Contents
NODE_GRP_NAME	C/I	Node group name or node ID
Value	R	Heat flux value

Parameter

```
!CFLUX
  ALL, 1.0E-3
!CFLUX, AMP=FUX1
  ALL, 1.0
```

#### 1.7.6.4.4 (4) !DFLUX (4-4)

Definition of distributed heat flux and internal heat generation given to surface of element

Parameter

AMP = Flux history table name (specified in !AMPLITUDE)

\*\* 2nd line or later \*\*

(2nd line) ELEMENT\_GRP\_NAME, LOAD\_type, Value

Paramater Name	Attributions	Contents
ELEMENT_GRP_NAME	C/I	Element group name or element ID
LOAD_type	C	Load type No.
Value	R	Heat flux value

Parameter

```
!DFLUX
  ALL, S1, 1.0
!DFLUX, AMP=FLUX2
  ALL, S0, 1.0
```

Load Parameters

Load Type No.	Applied Surface	Parameter
BF	Element overall	Calorific value
S1	Surface No. 1	Heat flux value
S3	Surface No. 2	Heat flux value
S4	Surface No. 3	Heat flux value
S5	Surface No. 4	Heat flux value
S6	Surface No. 5	Heat flux value
S2	Surface No. 6	Heat flux value
S3	Shell surface	Heat flux value

#### 1.7.6.4.5 (5) !SFLUX (4-5)

Definition of distributed heat flux by surface group

Parameter

AMP = Flux history table name (specified in !AMPLITUDE)

\*\* 2nd line or later \*\*

(2nd line) SURFACE\_GRP\_NAME, Value

Parameter Name	Attributions	Contents
SURFACE_GRP_NAME	C	Surface group name
Value	R	Heat flux value

Example of Use

```
!SFLUX
  SURF, 1.0
!SFLUX, AMP=FLUX3
  SURF, 1.0
```

#### 1.7.6.4.6 (6) !FILM (4-6)

Definition of heat transfer coefficient given to the boundary plane

Parameter

AMP1 = Heat transfer coefficient history table name (specified in !AMPLITUDE)

AMP2 = Ambient temperature history table name (specified in !AMPLITUDE)

\*\* 2nd line or later \*\*

(2nd line) ELEMENT\_GRP\_NAME, LOAD\_type, Value, Sink

Parameter Name	Attributions	Contents
ELEMENT_GRP_NAME	C/I	Element group name or element ID
LOAD_type	C	Load type No.
Value	R	Heat transfer coefficient
Sink	R	Ambient temperature

Example of Use

```
!FILM
  FSURF, F1, 1.0, 800.0
!FILM, AMP1=TFILM
  FSURF, F1, 1.0, 1.0
```

Load Parameters

Load Type No.	Applied Surface	Parameter
F1	Surface No. 1	Heat transfer coefficient and ambient temperature
F2	Surface No. 2	Heat transfer coefficient and ambient temperature
F3	Surface No. 3	Heat transfer coefficient and ambient temperature
F4	Surface No. 4	Heat transfer coefficient and ambient temperature
F5	Surface No. 5	Heat transfer coefficient and ambient temperature
F6	Surface No. 6	Heat transfer coefficient and ambient temperature
F0	Shell Surface	Heat transfer coefficient and ambient temperature

#### 1.7.6.4.7 (7) !SFILM (4-7)

Definition of heat transfer coefficient by surface group

Parameter

AMP1 = Heat transfer coefficient history table name (specified in !AMPLITUDE)  
AMP2 = Ambient temperature history table name (specified in !AMPLITUDE)

\*\* 2nd line or later \*\*

(2nd line) SURFACE\_GRP\_NAME, Value, Sink

Parameter Name	Attributions	Contents
SURFACE_GRP_NAME	C	Surface group name
Value	R	Heat Transfer Rate
Sink	R	Ambient Temperature

Example of Use

```
!SFILM
  SFSURF, 1.0, 800.0
!SFILM, AMP1=TSFILM, AMP2=TFILM
  SFSURF, 1.0, 1.0
```

#### 1.7.6.4.8 !RADIATE (4-8)

Definition of radiation factor given to boundary plane

Parameter

AMP1 = Radiation factor history table name (specified in !AMPLITUDE)  
AMP2 = Ambient temperature history table name (specified in !AMPLITUDE)

\*\* 2nd line or later \*\*

(2nd line) ELEMENT\_GRP\_NAME, LOAD\_type, Value, Sink

Parameter Name	Attributions	Contents
ELEMENT_GRP_NAME	C/I	Element group name or element ID
LOAD_type	C	Load type No.
Value	R	Radiation factor
Sink	R	Ambient temperature

Example of Use

```
!RADIATE
  RSURF, R1, 1.0E-9, 800.0
!RADIATE, AMP2=TRAD
  RSURF, R1, 1.0E-9, 1.0
```

Load Parameters

Load Type No.	Applied Surface	Parameter
R1	Surface No. 1	Radiation factor and ambient temperature
R2	Surface No. 2	Radiation factor and ambient temperature
R3	Surface No. 3	Radiation factor and ambient temperature
R4	Surface No. 4	Radiation factor and ambient temperature
R5	Surface No. 5	Radiation factor and ambient temperature
R6	Surface No. 6	Radiation factor and ambient temperature
R0	Shell Surface	Radiation factor and ambient temperature

#### 1.7.6.4.9 (9) !SRADIATE (4-9)

Definition of radiation factor by surface group

Parameter

AMP1 = Radiation factor history table name (specified in !AMPLITUDE)

AMP2 = Ambient temperature history table name (specified in !AMPLITUDE)

\*\* 2nd line or later \*\*

(2nd line) SURFACE\_GRP\_NAME, Value, Sink

Parameter Name	Attributions	Contents
SURFACE_GRP_NAME	C	Surface group name
Value	R	Radiation factor
Sink	R	Ambient temperature

Example of Use

```
!SRADIATE
  RSURF, 1.0E-9, 800.0
!SRADIATE, AMP2=TSRAD
  RSURF, 1.0E-9, 1.0
```

#### 1.7.6.4.10 (10) !WELD\_LINE (4-10)

Definition of weld line (Linear)

Parameter

N/A

\*\* 2nd line \*\*

(2nd line) I, U, Coef, V

Parameter Name	Attributions	Contents
I	R	Current
U	R	Voltage
Coef	R	Heat input coefficient
V	R	Movement speed of the welding torch

\*\* 3rd line \*\*

(3rd line) EGROUP, XYZ, C1, C2, H, tstart

Parameter Name	Attributions	Contents
EGROUP	C	Element group name for heat input
XYZ	I	Movement direction of welding torch (Degree of freedom No.)
C1	R	Starting point coordinates of welding torch
C2	R	Ending point coordinates of welding torch
H	R	Width of welding torch, inside which thermo energy inputted
tstart	R	Welding start time

#### 1.7.6.5 Control Data for Dynamic Analysis

### 1.7.6.5.1 (1) DYNAMIC

Dynamic analysis control

Time t for each !AMPLITUDE specified in !BOUNDARY, !CLOAD and !DLOAD must be started from 0.0.

Parameter

TYPE = LINEAR : Linear dynamic analysis  
 NONLINEAR : Nonlinear dynamic analysis

\*\* 2nd line or later \*\*

(2nd line) idx\_eqa , idx\_resp

Name	Attributions	Contents
idx_eqa	I	Solution of equation of motion (Direct time integration)(Default: 1) 1: Implicit method (Newmark- method) 11: Explicit method (Center difference method)
idx_resp	I	Analysis type (Default: 1) 1: Time history response analysis 2: Frequency response analysis (Not included)

idx\_resp=1 (Time history response analysis)

(3rd line) t\_start , t\_end , n\_step , t\_delta

Parameter Name	Attributions	Contents
t_start	R	Analysis start time (Default: 0.0), not used
t_end	R	Analysis end time (Default: 1.0), not used
n_step	I	Overall No. of steps (Default: 1)
t_delta	R	Time increment (Default: 1.0)

(4th line) gamma , beta

Parameter Name	Attributions	Contents
gamma	R	Parameter of Newmark- method (Default: 0.5)
beta	R	Parameter of Newmark- method (Default: 0.25)

(5th line) idx\_mas ,idx\_dmp , ray\_m ,ray\_k

Name	Attributions	Contents
idx_mas	I	Type of mass matrix (Default: 1) 1: Lumped mass matrix 2: Consistent mass matrix
idx_dmp	I	1: Rayleigh damping (Default: 1)
ray_m	R	Parameter Rm of Rayleigh damping (Default: 0.0)
ray_k	R	Parameter Rk of Rayleigh damping (Default: 0.0)

(6th line) nout , node\_monit\_1 , nout\_monit

Parameter Name	Attribu-tions	Contents
nout	I	not used
node_monit_1	I	Monitoring node ID (Global) or node group name
nout_monit	I	Results output interval of displacement monitoring(Default: 1)

Note: Regarding the information of the monitoring node specified in this line, the displacement is output to the file <dyna\_disp\_NID.txt>, where NID is the global ID of the monitoring node, and each line includes the step number, time of the step, NID, u1, u2, and u3 in this order. The velocity and acceleration are also output to <dyna\_velo\_NID.txt> and <dyna\_acce\_NID.txt>, respectively, in the same format as the displacement. The nodal strain is output to <dyna\_strain\_NID.txt> and each line includes the step number, time of the step, NID, e11, e22, e33, e12, e23, and e13 in this order. The nodal stress is output to <dyna\_stress\_NID.txt> and each line includes the step number, time of the step, NID, s11, s22, s33, s12, s23, s13, and s\_mises in this order. When monitoring nodes are specified by a node group, each of the files stated above is separately output for each node. When this output is specified, the kinetic energy, deformation energy and the overall energy of the overall analytic model will also be output to <dyna\_energy.txt>.

(7th line) iout\_list(1), iout\_list(2), iout\_list(3), iout\_list(4), iout\_list(5), iout\_list(6)

Parameter Name	Attribu-tions	Contents
iout_list(1)	I	Displacement output specification (Default: 0): 0: Not output, 1: Output
iout_list(2)	I	Velocity output specification (Default: 0): 0: Not output, 1: Output
iout_list(3)	I	Acceleration output specification (Default: 0): 0: Not output, 1: Output
iout_list(4)	I	Reaction force output specification (Default: 0): 0: Not output, 1: Output
iout_list(5)	I	Strain output specification (Default: 0): 1: Output2: Output (Node base)3: Output (Element base)
iout_list(6)	I	Stress output specification (Default: 0): 0: Not output (Element base and node base)1: Output2: Output (Node base)3: Output (Element base)

#### Example of Use

```
!DYNAMIC, TYPE=NONLINEAR
  1 , 1
  0.0 , 1.0 , 500 , 1.0000e-5
  0.5 , 0.25
  1 , 1 , 0.0 , 0.0
  100 , 55 , 1
  0 , 0 , 0 , 0 , 0
```

idx\_resp=2 (Frequency response analysis)

(3rd line) f\_start, f\_end, n\_freq, f\_disp

Parameter Name	Attributions	Contents
f_start	R	Minimum frequency
f_end	R	Maximum frequency
n_freq	I	Number of divisions for the frequency range
f_disp	R	Frequency to obtain displacement

(4th line) t\_start, t\_end

Parameter Name	Attributions	Contents
t_start	R	Analysis start time
t_end	R	Analysis end time

(5th line) idx\_mas , idx\_dmp , ray\_m , ray\_k

Parameter Name	Attributions	Contents
idx_mas	I	Type of mass matrix (Default: 1) 1: Lumped mass matrix
idx_dmp	I	1: Rayleigh damping (Default: 1)
ray_m	R	Parameter Rm of Rayleigh damping (Default: 0.0)
ray_k	R	Parameter Rk of Rayleigh damping (Default: 0.0)

(6th line) nout , vistype , nodeout

Parameter Name	Attributions	Contents
nout	I	Results output interval in time domain
vistype	I	Visuzalization type 1:Mode shapes 2:Time history results at f_disp
nodeout	I	Monitoring NODE ID in frequency domain

(7th line) iout\_list(1) , iout\_list(2) , iout\_list(3) , iout\_list(4) , iout\_list(5) , iout\_list(6)

Parameter Name	Attributions	Contents
iout_list(1)	I	Displacement output specification (Default: 0) 0: Not output, 1: Output
iout_list(2)	I	Velocity output specification (Default: 0) 0: Not output, 1: Output
iout_list(3)	I	Acceleration output specification (Default: 0) 0: Not output, 1: Output
iout_list(4)	I	not used
iout_list(5)	I	not used
iout_list(6)	I	not used

#### Example of Use

```
!DYNAMIC
 11 , 2
 14000, 16000, 20, 15000.0
 0.0 , 6.6e-5
 1, 1, 0.0 , 7.2E-7
 10, 2, 1
 1, 1, 1, 1, 1
```

#### 1.7.6.5.2 (2) !VELOCITY (5-2)

Definition of velocity boundary conditions

Parameter

```
TYPE = INITIAL (Initial velocity boundary conditions)
      = TRANSIT (Time history velocity boundary conditions
                  specified in !AMPLITUDE; Default)
AMP = Time function name (specified in !AMPLITUDE)
      Provides the relationship between time t and factor f(t) in !AMPLITUDE.
```

The time multiplied by factor  $f(t)$  to the following value becomes the restrained value of that time (when not specified: time and factor relationship becomes  $f(t) = 1.0$ ).

\*\* 2nd line or later \*\*

(2nd line) NODE\_ID, DOF\_idS, DOF\_idE, Value

Parameter Name	Attributions	Contents
NODE_ID	I/C	Node ID or node group name
DOF_idS	I	Start No. of restricted degree of freedom
DOF_idE	I	End No. of restricted degree of freedom
Value	R	Restricted value (Default: 0)

#### Example of Use

```
!VELOCITY, TYPE=TRANSIT, AMP=AMP1
  1, 1, 1, 0.0
  ALL, 3, 3
  * Restricted value is 0.0
!VELOCITY, TYPE=INITIAL
  1, 3, 3, 1.0
  2, 3, 3, 1.0
  3, 3, 3, 1.0
```

Note: The velocity boundary conditions are different than the displacement boundary conditions, and the multiple degrees of freedom can not be defined collectively. Therefore, the same number must be used for DOF\_idS and DOF\_idE. When the TYPE is INITIAL, AMP becomes invalid.

#### 1.7.6.5.3 (3) !ACCELERATION (5-3)

Definition of acceleration boundary conditions

Parameter

TYPE = INITIAL (Initial acceleration boundary conditions)  
   = TRANSIT ((Time history acceleration boundary conditions  
     specified in AMPLITUDE; Default))  
 AMP = Time function name (specified in !AMPLITUDE)  
   Provides the relationship between time t and factor  $f(t)$  in !AMPLITUDE.  
   The time multiplied by factor  $f(t)$  to the following Value  
   becomes the restrained value of that time (when not specified:  
   time and factor relationship becomes  $f(t) = 1.0$ ).

\*\* 2nd line or later \*\*

(2nd line) NODE\_ID, DOF\_idS, DOF\_idE, Value

Parameter Name	Attributions	Contents
NODE_ID	I/C	Node ID or node group name
DOF_idS	I	Start No. of restricted degree of freedom
DOF_idE	I	End No. of restricted degree of freedom
Value	R	Restricted value (Default: 0)

#### Example of Use

```
!ACCELERATION, TYPE=TRANSIT, AMP=AMP1
  1, 1, 3, 0.0
```

```

ALL, 3, 3
i* Restricted value is 0.0
!ACCELERATION, TYPE=INITIAL
1, 3, 3, 1.0
2, 3, 3, 1.0
3, 3, 3, 1.0

```

Note: The acceleration boundary conditions are different than the displacement boundary conditions, and the multiple degrees of freedom can not be defined collectively. Therefore, the same number must be used for DOF\_idS and DOF\_idE.

When the TYPE is INITIAL, AMP becomes invalid.

#### 1.7.6.5.4 (4) !COUPLE (5-4)

Definition of coupled surface (Used only in coupled analysis)

Parameter

TYPE = 1: One-way coupled (FrontISTR starts from receiving data)  
 2: One-way coupled (FrontISTR starts from sending data)  
 3: Staggered two-way coupled (FrontISTR starts from receiving data)  
 4: Staggered Two-way coupled (FrontISTR starts from sending data)  
 5: Iterative partitioned two-way coupled (FrontISTR starts from receiving data)  
 6: Iterative partitioned two-way coupled (FrontISTR starts from sending data)

ISTEP = Step No.

From the beginning of analysis to the step specified here, a linearly increasing function from 0 to 1 is multiplied to the input fluid traction.

After this step, the input fluid traction is directly applied.

WINDOW => 0: Multiply window function(\*) to input fluid traction

(\*)  $\frac{1}{2}(1 - \cos \frac{2\pi i}{N})$ ,  $i$ : current step,  $N$ : no. of steps of current analysis

\*\* 2nd line or later \*\*

(2nd line) COUPLING\_SURFACE\_ID

Parameter Name	Attributions	Contents
SURFACE_ID	C	Surface group name

Example of Use

```

!COUPLE , TYPE=1
SCOUPLE1
SCOUPLE2

```

#### 1.7.6.5.5 (5) !EIGENREAD (5-5)

Controlling the input file for frequency response analysis

Parameter

N/A

\*\* 2nd line or later \*\*

Parameter Name	Attributions	Contents
eigenlog_filename	C	The name of eigenvalue analysis log

(3rd line) start\_mode, end\_mode

Parameter Name	Attributions	Contents
start_mode	I	lowest mode to be used in frequency response analysis
end_mode	I	highest mode to be used in frequency response analysis

Example of Use

```
!EIGENREAD
eigen_0.log
1, 5
```

#### 1.7.6.5.6 (6) !FLOAD (5-6)

Defining external forces applied in frequency response analysis

Parameter

LOAD CASE = (1: Real part , 2: Imaginary part)

\*\* 2nd line or later \*\*

(2nd line) NODE\_ID, DOF\_id, Value

Parameter Name	Attributions	Contents
NODE_ID	I/C	Node ID, node group name or surface group name
DOF_id	I	Degree of freedom No.
Value	R	Load value

Example of Use

```
!FLOAD, LOAD CASE=2
_PickedSet5, 2, 1.0
```

#### 1.7.6.6 Solver Control Data

##### 1.7.6.6.1 (1) !SOLVER (6-1)

Control of solver

Mandatory control data

##### 1.7.6.6.2 Parameter

METHOD = Method (CG, BiCGSTAB, GMRES, GPBiCG, DIRECT, DIRECTmkl, MUMPS)  
 DIRECT: Direct method other than contact analysis (serial processing only) (curr  
 DIRECTmkl: Direct method by Intel MKL  
 MUMPS : Direct method by MUMPS  
 When any of direct methods is selected, the data lines will be disregarded.  
 Thread-parallel computation by OpenMP is available in iterative methods  
 for 3D problems .

PRECOND = Preconditioner for iterative methods (1, 2, 3, 5, 10, 11, 12)  
 1, 2 : (Block) SSOR (with multi-color ordering only for 3D problems)  
 3 : (Block) Diagonal Scaling  
 5 : AMG by multigrid preconditioner package ML  
 10 : Block ILU(0)  
 11 : Block ILU(1)  
 12 : Block ILU(2)  
 10, 11 and 12 are available only in 3D problems .

In thread-parallel computation , SSOR, Diagonal Scaling or ML is recommended.

**ITERLOG** = Whether solver convergence history is output (YES/NO) (Default: NO)

**TIMELOG** = Whether solver computation time is output (YES/NO/VERBOSE) (Default: NO)

**USEJAD** = Whether matrix ordering optimized for vector processors are performed (YES/NO) (Default: NO)  
Valid only in 3D problems with iterative solvers.

**SCALING** = Whether matrix is scaled so that each diagonal element becomes 1 (YES/NO) (Default: NO)  
Valid only in 3D problems with iterative solvers.

**DUMPTYPE** = Type of matrix dumping (NONE, MM, CSR, BSR) (Mainly for debugging)  
NONE : no dumping (Default)  
MM : matrix is dumped in Matrix Market format  
CSR : matrix is dumped in Compressed Sparse Row (CSR) format  
BSR : matrix is dumped in Blocked CSR format

**DUMPEXIT** = Whether the program terminates right after matrix dumping (YES/NO) (Default: NO)

**MPCMETHOD** = Method for multipoint constraints  
1: Penalty method (Default for direct methods)  
2: MPC-CG method  
3: Explicit master-slave elimination (Default for iterative methods)

**ESTCOND** = Frequency of condition number estimation (experimental)  
Estimation is performed at every specified number of iterations and at the last iteration. No estimation when 0 is specified.

**METHOD2** = Secondary method (BiCGSTAB, GMRES, GPBiCG) (experimental)  
Valid only when CG is specified as **METHOD**.  
When specified , the method is switched and solution continues when CG diverged.  
All the other parameters and data lines are shared with the CG method.

\*\* 2nd line or later \*\*

(2nd line) NITER, iterPREmax, NREST, NCOLOR\_IN, RECYCLEPRE

Parameter Name	Attributions	Contents
NITER	I	No. of iterations (Default: 100)
iterPREmax	I	No. of iteration of preconditioning based on Additive Schwarz(Default: 1)(recommended value : 1 (2 might be efficient in some parallel computation))
NREST	I	No. of Krylov subspaces (Default: 10)(Valid only when GMRES is selected as the solution)
NCOLOR_IN	I	No. of Colors for Multi-Color ordering (Default: 10)(Valid only when no. of OpenMP threads >= 2)
RECYCLEPRE	I	No. of recycling set-up info for preconditioning (Default: 3)(Valid only in nonlinear analyses)

(3rd line) RESID, SIGMA\_DIAG, SIGMA

Parameter Name	Attribu-tions	Contents
RESID	R	Truncation error (Default: 1.0e-8)
SIGMA_DIAG	R	Scale factor for diagonal elements when computing preconditioning matrix (Default: 1.0)(When divide-by-zero or divergence occurs with ILU preconditioning, convergence might be obtained by setting number greater than 1.0)
SIGMA	R	Not used (Default: 0.0)

In case of PRECOND=5 (Optional)

When any other value is specified for PRECOND, the 4th line will be disregarded.

(4th line) ML\_CoarseSolver , ML\_Smooth , ML\_MGCycle, ML\_MaxLevels , ML\_CoarseningScheme , ML\_N

Parameter Name	Attribu-tions	Contents
ML_CoarseSolver	I	Coarse solver of ML (1: smoother, 2: KLU (serial direct solver), 3: MUMPS (parallel direct solver)) (Default: 1) (recommended value : 3 or 2 for stiff problems, 1 for other problems)
ML_Smooth	I	Smoothen of ML (1: Chebyshev, 2: SymBlockGaussSeidel, 3: Jacobi) (Default: 1) (recommended value : 1)
ML_MGCycle	I	Multigrid cycle of ML (1: V-cycle, 2: W-cycle, 3: Full-V-cycle) (Default: 1) (recommended value : 2 for stiff problems, 1 for other problems)
ML_MaxLevels	I	Max No. of levels of ML (Default: 10) (recommended value : 2 (or 3 when memory is not sufficient) with direct coarse solver for very stiff problems, 10 for other problems)
ML_CoarseningScheme	I	Coarsening scheme of ML (1: UncoupledMIS, 2: METIS, 3: ParMETIS, 4: Zoltan, 5: DD) (Default: 1) (recommended value : 1 or 5)
ML_NumSweep	I	No. of smoother sweeps of ML (polynomial degree for Chebyshev) (Default: 2) (recommended value : 2 for Chebyshev, 1 for SymBlockGaussSeidel)

#### Example of Use

Use CG with SSOR preconditioning, and set No. of iteration to 10000 and truncation error to 1.0e-8

```
!SOLVER, METHOD=CG, PRECOND=1, ITERLOG=YES, TIMELOG=YES
10000, 1
1.0e-8, 1.0, 0.0
```

Use GMRES with SSOR preconditioning, and set No. of Krylov subspace to 40 and No. of colors for Multi-Color ordering to 100

```
!SOLVER, METHOD=GMRES, PRECOND=1, ITERLOG=YES, TIMELOG=YES
10000, 1, 40, 100
1.0e-8, 1.0, 0.0
```

Use CG with ILU(0) preconditioning, and set scale factor for diagonal elements when computing preconditioning matrix to 1.1

```
!SOLVER, METHOD=CG, PRECOND=10, ITERLOG=YES, TIMELOG=YES
10000, 1
1.0e-8, 1.1, 0.0
```

Use CG with AMG preconditioning by ML

```

!SOLVER, METHOD=CG, PRECOND=5, ITERLOG=YES, TIMELOG=YES
 10000, 1
 1.0e-8, 1.0, 0.0

```

Use CG with AMG preconditioning by ML, and set coarse solver to MUMPS (for stiff problems)

```

!SOLVER, METHOD=CG, PRECOND=5, ITERLOG=YES, TIMELOG=YES
 10000, 1
 1.0e-8, 1.0, 0.0
 3

```

Use CG with AMG preconditioning by ML, and set multigrid cycle to W-cycle (for stiff problems)

```

!SOLVER, METHOD=CG, PRECOND=5, ITERLOG=YES, TIMELOG=YES
 10000, 1
 1.0e-8, 1.0, 0.0
 1, 1, 2

```

Use CG with AMG preconditioning by ML, and set coarse solver to MUMPS and max No. of levels to 2 (for very stiff problems)

```

!SOLVER, METHOD=CG, PRECOND=5, ITERLOG=YES, TIMELOG=YES
 10000, 1
 1.0e-8, 1.0, 0.0
 3, 1, 2

```

### 1.7.6.7 Post Process (Visualization) Control Data

#### 1.7.6.7.1 (1) !VISUAL (P1-0)

Specifies the visualization method.

METHOD = PSR	:	Surface rendering
visual_start_step	:	Specification of time step number which starts the visualization (Default: 1)
visual_end_step	:	Specification of time step number which ends the visualization (Default: All)
visual_interval_step	:	Specification of time step interval which performs the visualization (Default: 1)

#### 1.7.6.7.2 (2) !surface\_num, !surface, !surface\_style (P1-1 - 3)

**!surface\_num** (P1-1)

No. of surfaces in one surface rendering

Ex.: There are four surfaces in Figure 7.4.1, which includes two isosurfaces pressure = 1000.0 and pressure = -1000.0, and two cut end plane surfaces z = -1.0 and z = 1.0.

**Figure 7.4.1: Example of surface\_num Setting**

**!surface** (P1-2)

Sets the contents of the surface.

Ex: Then contents of the four surface in Figure 7.4.2 are as follows.

Figure 7.4.2: Example of Surface Setting

```

!surface_num = 2
!SURFACE
!surface_style = 2
!data_comp_name = press
!iso_value = 1000.0

```

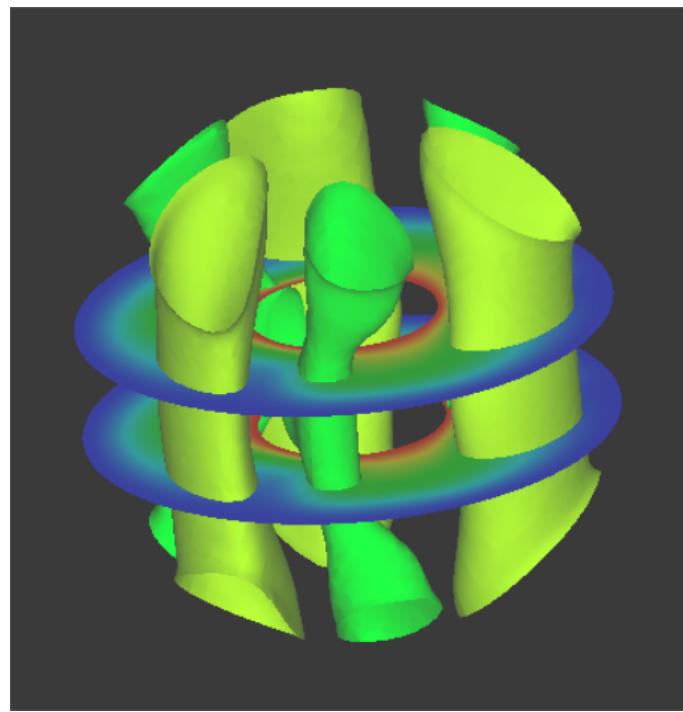


Figure 19: Example of surface\_num Setting

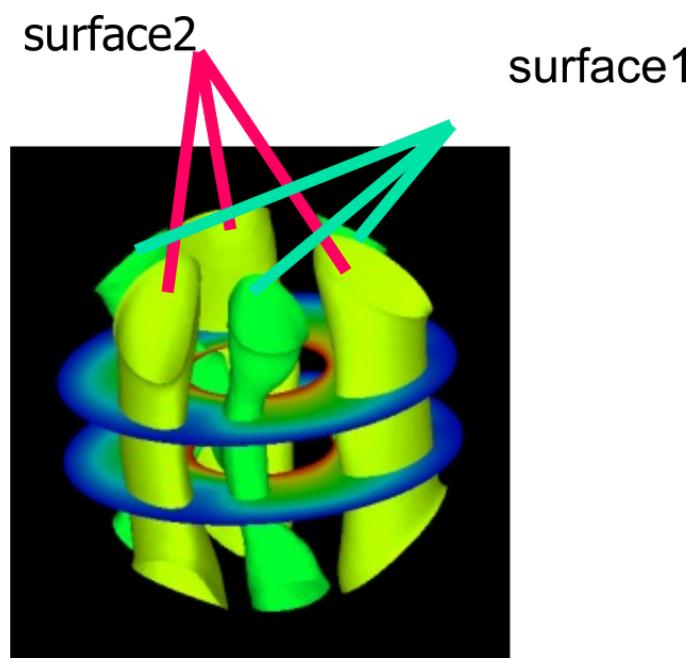


Figure 20: Example of Surface Setting

```

!display_method = 4
!specified_color = 0.45
!output_type = BMP
!SURFACE
!surface_style = 2
!data_comp_name = press
!iso_value = -1000.0
!display_method = 4
!specified_color = 0.67

```

#### **!surface\_style (P1-3)**

Specifies the style of the surface.

1. Boundary plane
2. Isosurface
3. Arbitrary quadric surface  $\text{coef}[1]x^2 + \text{coef}[2]y^2 + \text{coef}[3]z^2 + \text{coef}[4]xy + \text{coef}[5]xz + \text{coef}[6]yz + \text{coef}[7]x + \text{coef}[8]y + \text{coef}[9]z + \text{coef}[10]=0$

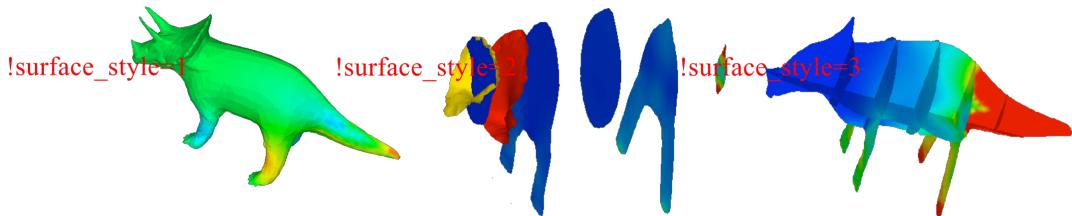


Figure 21: Example of surface\_style Setting

**Figure 7.4.3: Example of surface\_style Setting**

#### **1.7.6.7.3 (3) !display\_method (P1-4)**

Display method (Default: 1)

1. Color code display
2. Boundary line display
3. Color code and boundary line display
4. Display of 1 specified color
5. Isopleth line display by classification of color

**Figure 7.4.4: Example of display\_method Setting**

#### **1.7.6.8 (2) !color\_comp\_name, !color\_comp, !color\_subcomp (P1-5, P1-7, P1-8)**

Specifies the selections for the color map from the physical values. Provides the names to the necessary physical values and the degree of freedom numbers. Accordingly, the names will be entered for the structure node\_label(:) and nn\_dof(:) of the results data.

Then you can define which one you hope to map into color by

**!color\_comp\_name** (Character string, default: 1st parameter)

Example

```

!color_comp_name = pressure
In static analysis;           = DISPLACEMENT : Specification
                                         of the results displacement data
                                         = STRAIN      : Specification of strain data
                                         = STRESS      : Specification of stress data
In heat transfer analysis;   = TEMPERATURE : Specification
                                         of the results temperature data

```

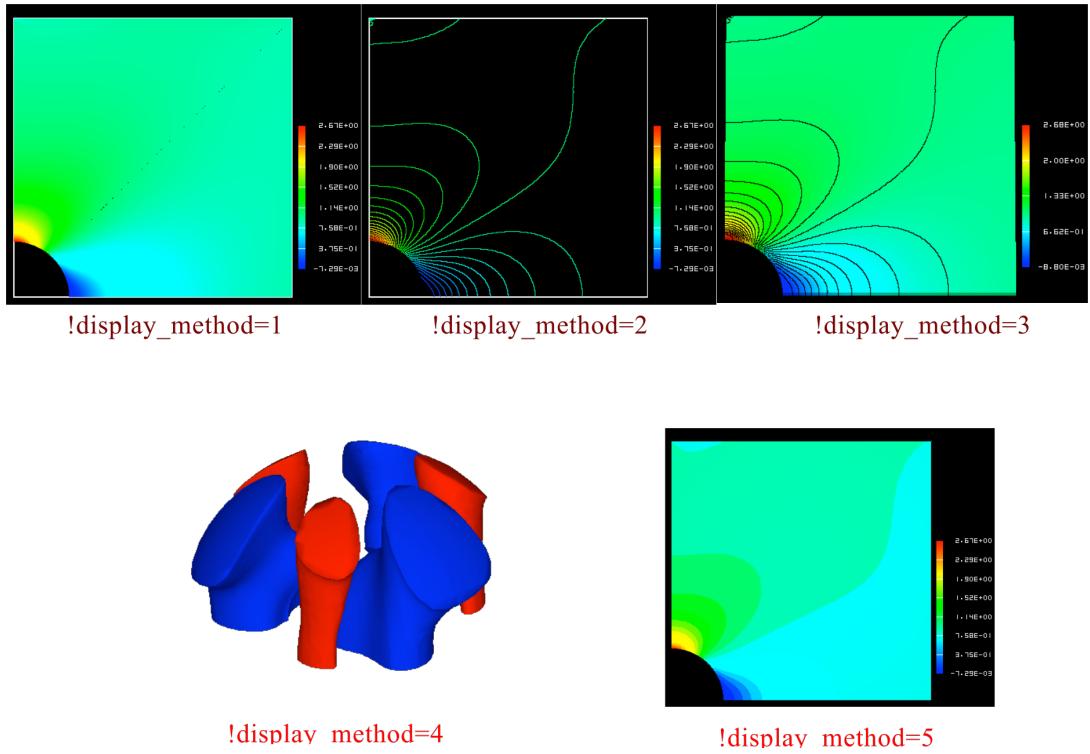


Figure 22: Example of display\_method Setting

**!color\_comp** (Integer, default: 0)

Physical value ID number (Integers above 0)

Example

**!color\_comp = 2**

This is the specification of the ID number and component name of the results data type; however, this is not included.

**!color\_subcomp** (Integer, default: 0)

When the physical value is 1 degree of freedom or more like the vector quantity, it's the number of the degree of freedom.

Example:

**!color\_subcomp = 0**

When **!color\_comp\_name=DISPLACEMENT** is specified

1: X Component , 2: Y Component , 3: Z Component

When **!color\_comp\_name=STRAIN** is specified

1:  $\epsilon_x$  , 2:  $\epsilon_y$  , 3:  $\epsilon_z$   
4:  $\epsilon_{xy}$  , 5:  $\epsilon_{yz}$  , 6:  $\epsilon_{zx}$

When **!color\_comp\_name=STRESS** is specified

1:  $\sigma_x$  , 2:  $\sigma_y$  , u:  $\sigma_z$   
4:  $\tau_{xy}$  , 5:  $\tau_{yz}$  , 6:  $\tau_{zx}$

When **!color\_comp\_name=TEMPERATURE** is specified

1: Temperature

In the structural analysis, for example;

Physical Value	Displacement	Strain	Stress
No. of degrees of freedom	3	6	7

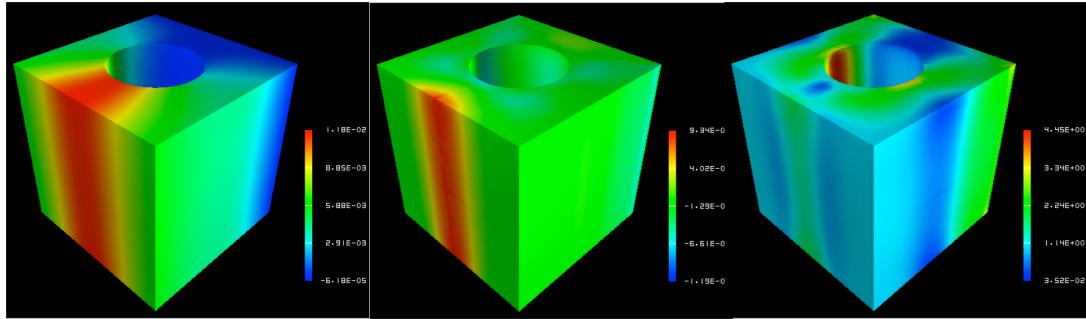


Figure 23: Example of color\_comp, color\_subcomp and color\_comp\_name Setting

**Figure 7.4.5: Example of color\_comp, color\_subcomp and color\_comp\_name Setting**

#### 1.7.6.8.1 (5) !isoline\_number, !isoline\_color (P1-9, P2-22)

When display\_method=2,3 or 5

```
!isoline_number = 30
!isoline_color = 0.0, 0.0, 0.0
```

```
!isoline_number = 10
!isoline_color = 1.0, 0.0, 0.0
```

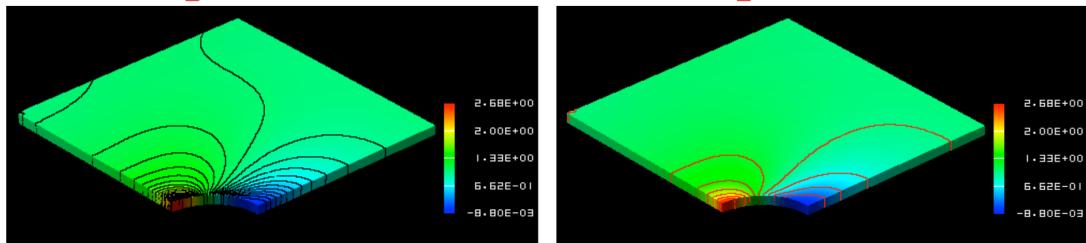


Figure 24: Example of isoline\_number and isoline\_color Setting

**Figure 7.4.6: Example of isoline\_number and isoline\_color Setting**

#### 1.7.6.8.2 (6) !initial\_style, !deform\_style (P1-15, P1-16)

Specifies the display style of the initial shape and the deformed shape.

0. Not specified
1. Solid line mesh (Displayed in blue if not specified)
2. Gray filled pattern
3. Shading (Let the physical attributions respond to the color)
4. Dotted line mesh (Displayed in blue if not specified)

#### (7) !deform\_scale (P1-14)

Specifies the displacement scale when displaying deformation.

Default: Auto

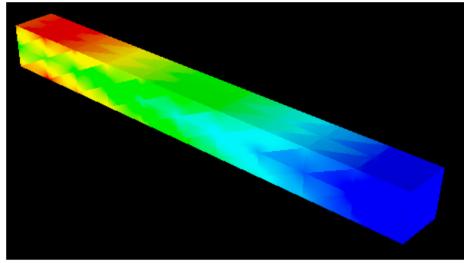
standard\_scale = 0.1 \* sqrt(x\_range2 + y\_range2 + z\_range2) / max\_deform

**Figure 7.4.7: Example of display\_styles Setting**

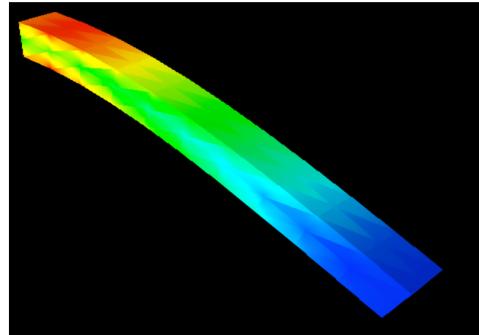
**Figure 7.4.8: Example of deform\_scale Setting**

#### (8) !output\_type (P1-19)

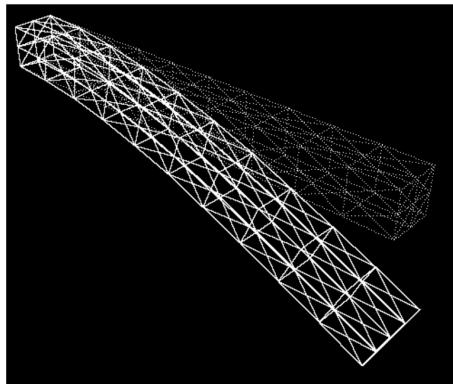
Specifies the type of output file. (Default: AVS)



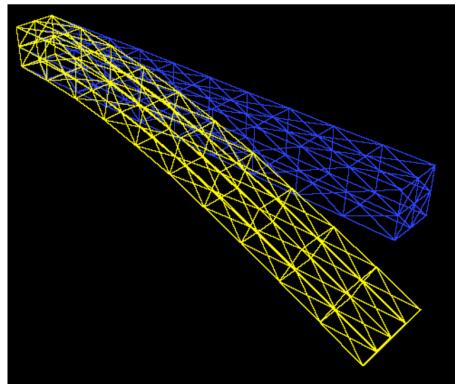
```
!initial_style=2  
!deform_style = 0
```



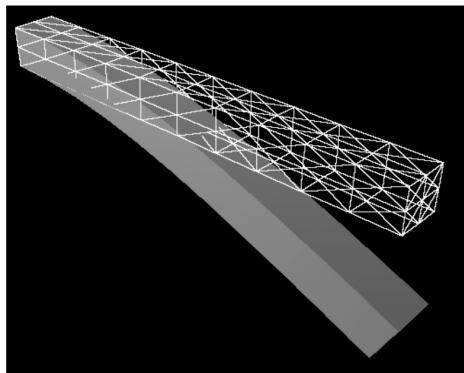
```
!initial_style=0  
!deform_style = 2
```



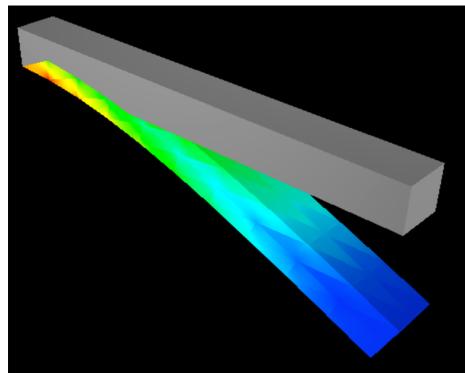
```
!initial_style=4  
!deform_style = 1  
!initial_line_color = 1.0, 1.0, 1.0
```



```
!initial_style=1  
!deform_style = 1  NASTRAN style  
!initial_line_color = default
```

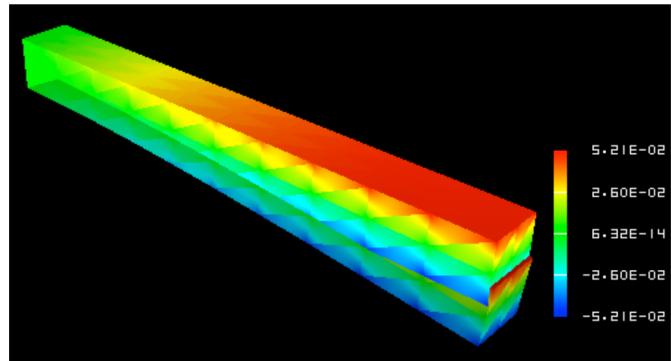


```
!initial_style=1  
!deform_style = 2
```

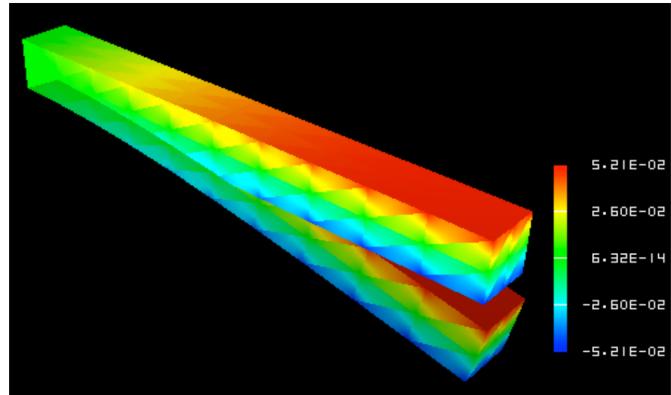


```
!initial_style=2  
!deform_style = 3
```

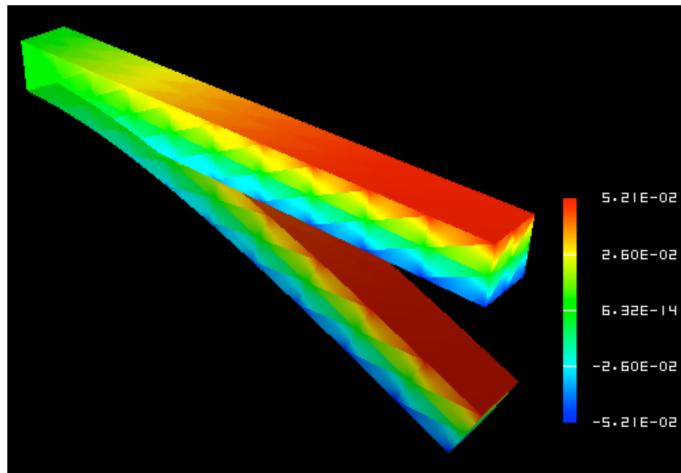
Figure 25: Example of Display Styles Setting



`!deform_scale=1.0`



`!deform_scale=2.0`



`!deform_scale=4.0`

Figure 26: Example of deform\_scale Setting

AVS	: UCD data for AVS (only on object surface)
BMP	: Image data (BMP format)
VTK	: VTK data for ParaView
COMPLETE_AVIS	: UCD data for AVS
COMPLETE_REORDER_AVIS	: Rearranges the node and element ID in the UCD data for AVS
SEPARATE_COMPLETE_AVIS	: UCD data for AVS for each decomposed domain
COMPLETE_MICROAVS	: Outputs the physical values in the scalar in the UCD data for AVS
BIN_COMPLETE_AVIS	: Outputs COMPLETE_AVIS in binary format
FSTR_FEMAP_NEUTRAL	: Neutral file for FEMAP

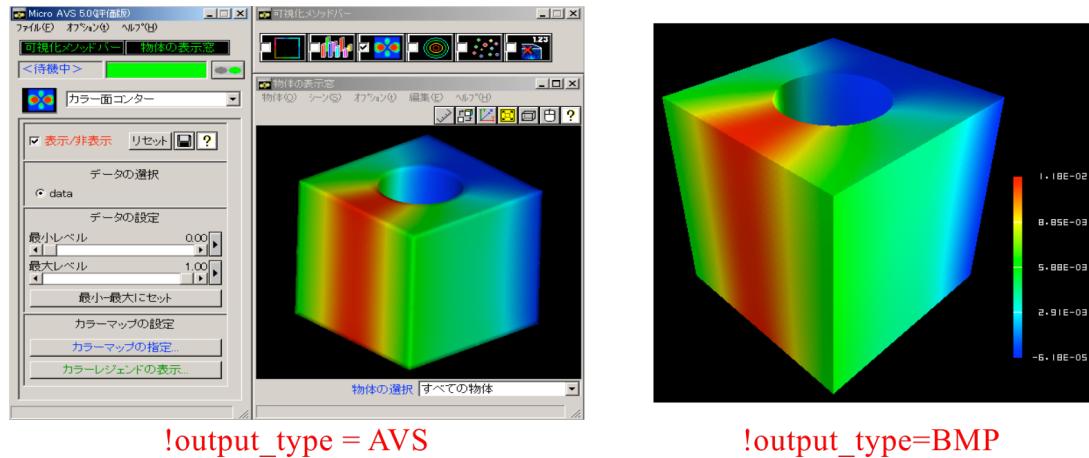


Figure 27: Example of output\_type

Figure 7.4.9: Example of output\_type

#### 1.7.6.8.3 (9) !x\_resolution, !y\_resolution (P2-1, P2-2)

Specifies the resolution when output\_type=BMP

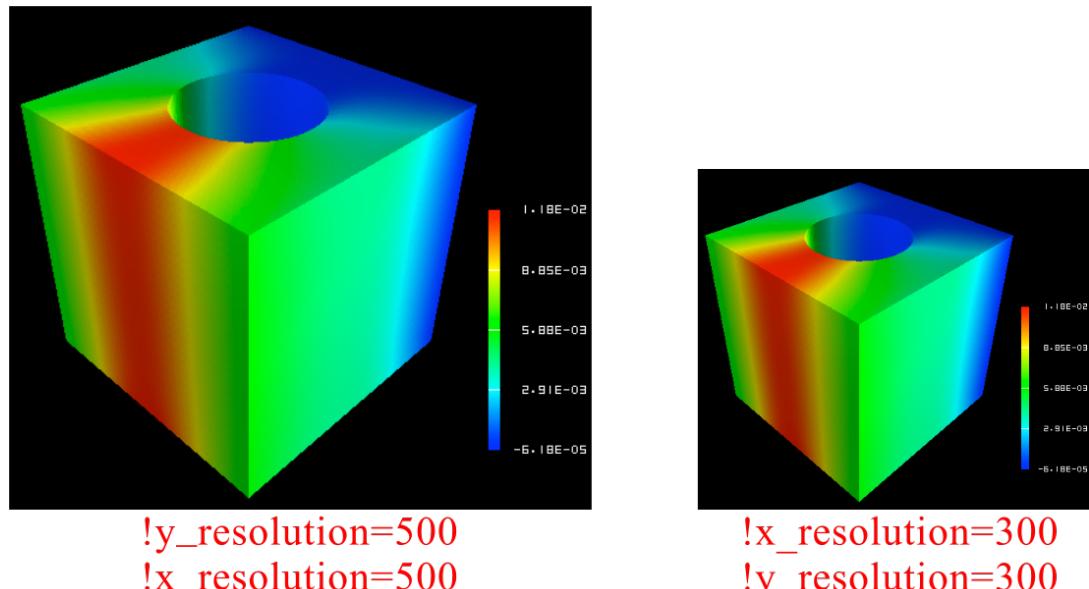


Figure 28: Example of x resolution and y resolution Setting

Figure 7.4.10: Example of x resolution and y resolution Setting

#### 1.7.6.8.4 (10) !viewpoint, !look\_at\_point, !up\_direction (P2-5, P2-6, P2-7)

viewpoint

Specifies the viewpoint position by coordinates.

Default:  $x = (\text{xmin} + \text{xmax})/2.0$ ,  $y = \text{ymin} + 1.5$  ( $\text{ymax} - \text{ymin}$ ),  $z = \text{zmin} + 1.5$  ( $\text{zmax} - \text{zmin}$ )

#### **look\_at\_point**

Specifies the look at point position.

Default: Center of data

#### **up\_direction**

Specifies the view frame in viewpoint, look\_at\_point and up\_direction.

default: 0.0 0.0 1.0

View coordinate frame

- Origin: look\_at\_point
- Z-axis: viewpoint - look\_at\_point
- X-axis: up\_direction  $\times$  z axis

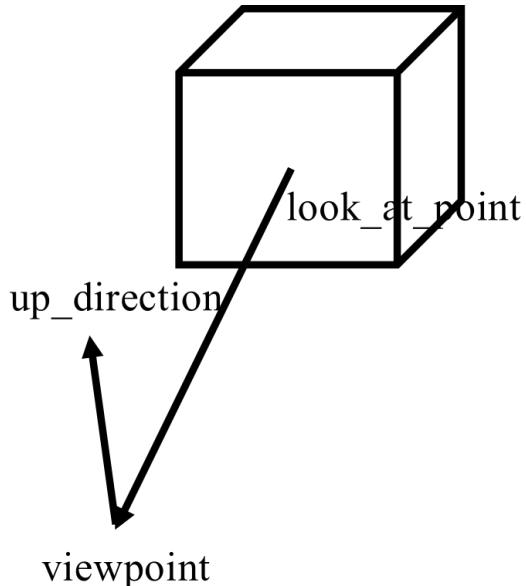


Figure 29: View Frame Determination Method

Figure 7.4.11: View Frame Determination Method

Figure 7.4.12: Example of !viewpoint, look\_at\_point and up\_direction Setting

#### **1.7.6.8.5 (11) !ambient\_coef !diffuse\_coef !specular\_coef (P2-8 P2-9 P2-10)**

Coefficient setting of lighting model

When the ambient\_coef is increased, information on the 3D depth direction is impaired.

#### **1.7.6.8.6 (12) ‘!color\_mapping\_bar\_on’ ‘!scale\_marking\_on’ ‘!num\_of\_scales’ (P2-16 P2-17 P2-18)**

---

**!color\_mapping\_bar\_on**

Specifies whether to display the color mapping bar. 0: off 1: on (Default: 0)

**!scale\_marking\_on**

set the memory status of color\_mapping\_bar0: off 1: on (default: 0)

**!num\_of\_scales**

Specifies the number of memory. (default: 3)

---

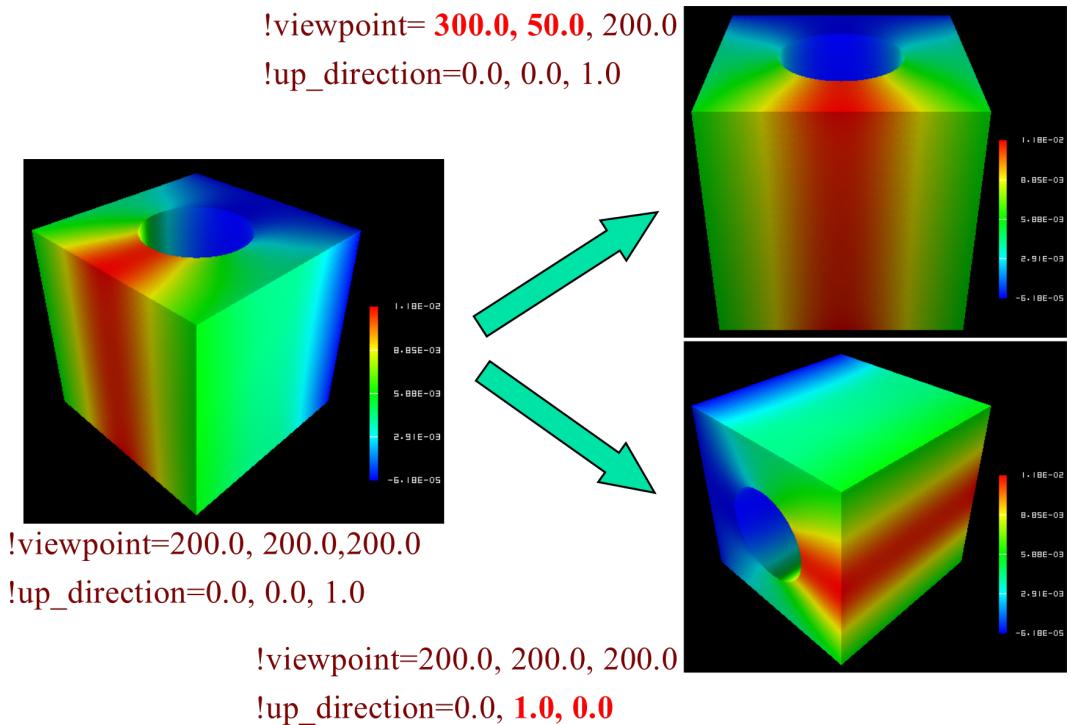


Figure 30: Example of !viewpoint, look\_at\_point and up\_direction Setting

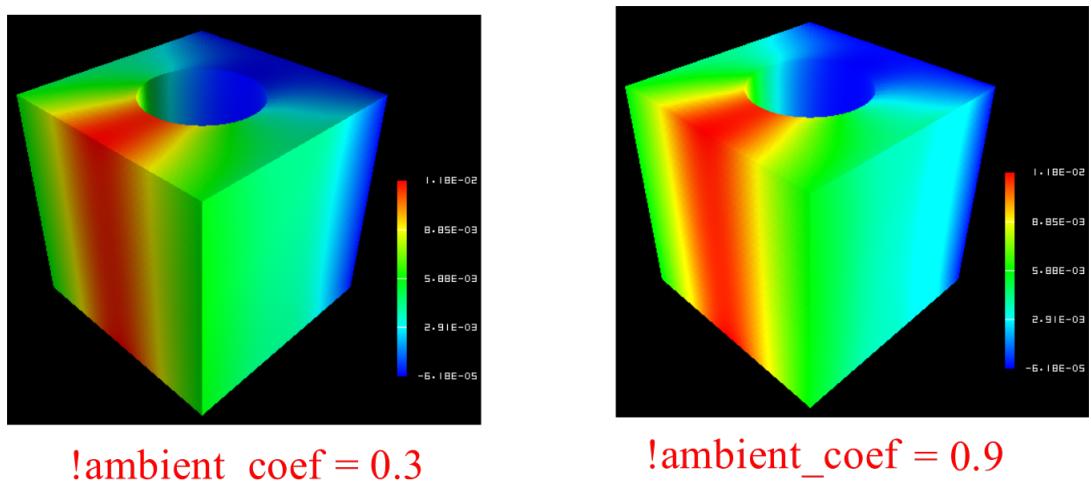


Figure 31: Example of Lighting Model Parameter Setting

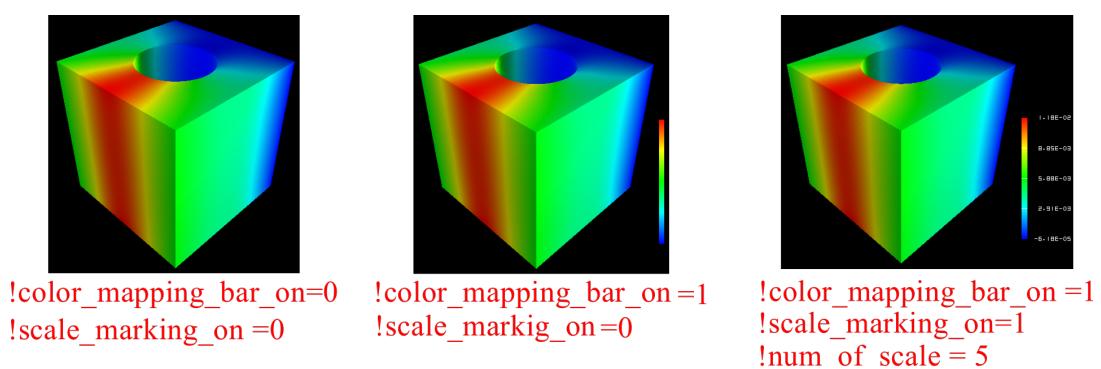


Figure 32: Example of Color Mapping Bar Display

#### 1.7.6.8.7 (13) !font\_size !font\_color !background\_color (P2-19 P2-20 P2-21)

Specifies the background color and character font.

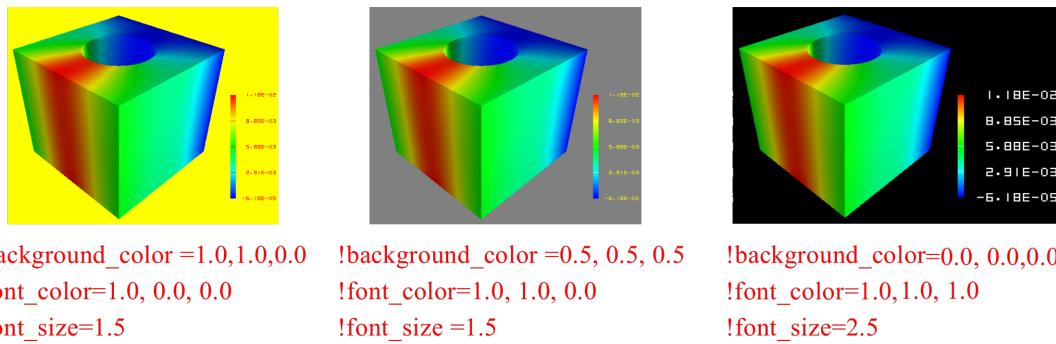


Figure 33: Example of Background and Font Setting

#### Figure 7.4.15: Example of Background and Font Setting

#### 1.7.6.8.8 (14) !data\_comp\_name, !data\_comp, !data\_subcomp (P3-1, P3-3, P3-4)

Specifies the physical values of the isosurface to be visualized when surface\_style=2.

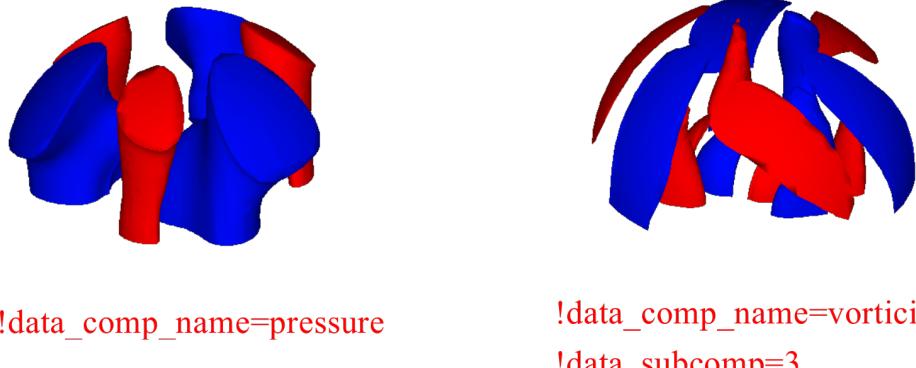


Figure 34: Example of data\_comp, data\_subcomp and data\_comp\_name Setting

#### Figure 7.4.16: Example of data\_comp, data\_subcomp and data\_comp\_name Setting

#### 1.7.6.8.9 (15) !method (P4-1)

When specifying the surfaces and cut end, specifies the setting method of the surface.

```
!surface_num =2
!surface
!surface_style=3
!method=5
!coef=0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 1.0, -0.35
!color_comp_name = temperature
```

#### Figure 7.4.17: Example of Setting Method

Accordingly, the cut end of the plane surface  $z = 0.35$  and  $z = -0.35$  will be visualized. ## User Subroutines

An interface is provided for users to expand the FrontISTR functions by programming. These interfaces are basically FORTRAN subroutines which include the subroutine header, and are a description of the I/O parameters and the declaration statement of these parameters. The main portion of the routine must be written by the user.

FrontISTR provides the following user subroutine interfaces.

### 1.7.7 Input of User Defined Material

When using user defined materials, up to a maximum of 100 material constants defined by the user can be used. As shown in the following, up to 10 values per line and a maximum of 10 lines of material constants can be input in the control data file.

#### 2nd line - 10th line maximaum

v1, v2, v3, v4, v5, v6, v7, v8, v9, v10

### 1.7.8 Subroutine regarding Elastoplasticity Deformation (**uyield.f90**)

The subroutines are provided in order to calculate the elastoplasticity stiffness matrix and stress return mapping. When using the user defined yield function, first, it is necessary to set the !PLASTIC, TYPE=USER in the input file, input the required material constants and then, create the subroutines uElastoPlasticMatrix and uBackwardEuler.

#### 1.7.8.1 (1) Calculation subroutines of elastoplasticity stiffness matrix

```
subroutine uElastoPlasticMatrix( matl, stress, istat, fstat, D )
  REAL(KIND=kreal), INTENT(IN) :: matl(:)
  REAL(KIND=kreal), INTENT(IN) :: stress(6)
  INTEGER, INTENT(IN) :: istat
  REAL(KIND=kreal), INTENT(IN) :: fstat(:)
  REAL(KIND=kreal), INTENT(OUT) :: D(:, :)
```

- matl: Array to save the material constants (100 max)
- stress: 2nd Piola-Kirchhoff stress
- istat: Yield state (0: not yielded; 1: yielded)
- fstat: State variable, fstat(1) = plastic strain, fstat(2:7) = back stress (while moving or complex hardening)
- D: Elastoplasticity matrix

#### 1.7.8.2 (2) Return mapping calculation subroutine of stress

```
subroutine uBackwardEuler ( matl, stress, istat, fstat )
  REAL(KIND=kreal), INTENT(IN) :: matl(:)
  REAL(KIND=kreal), INTENT(INOUT) :: stress(6)
  INTEGER, INTENT(INOUT) :: istat
  REAL(KIND=kreal), INTENT(IN) :: fstat(:)
```

- matl: Array to save the material constants (100 max)
- stress: 2nd Piola-Kirchhoff stress acquired by assuming trial stress elastic deformation
- istat: Yield state (0: not yielded; 1: yielded)
- fstat: State variable, fstat(1) = plastic strain, fstat(2:7) = back stress (while moving or complex hardening)

### 1.7.9 Subroutine regarding Elastic Deformation (**uelastic.f90**)

The subroutines are provided in order to perform update calculations of the elastic stiffness matrix and stress of the elasticity and hyperelasticity problems. When using the user elasticity, or a hyperelasticity constitutive equation, first, it is necessary to set the !ELASTIC, TYPE=USER or the !HYPERELASTIC, TYPE=USER in the input file, input the required material constants and then, create the subroutines uElasticMatrix and uElasticUpdate.

#### 1.7.9.1 (1) Calculation subroutine of elastic stiffness matrix

```
subroutine uElasticMatrix( matl, strain, D )
  REAL(KIND=kreal), INTENT(IN) :: matl(:)
  REAL(KIND=kreal), INTENT(IN) :: strain(6)
  REAL(KIND=kreal), INTENT(OUT) :: D(6,6)
```

- matl: Array to save the material constants (100 max)
- strain: Green-Lagrange strain
- D: Elastic matrix

### 1.7.9.2 (2) Calculation subroutine of stress

```
subroutine uElasticUpdate ( matl, strain , stress )
  REAL(KIND=kreal), INTENT(IN) :: matl(:)
  REAL(KIND=kreal), INTENT(IN) :: strain(6)
  REAL(KIND=kreal), INTENT(OUT) :: stress(6)
```

- matl: Array to save the material constants (100 max)
- strain: Green-Lagrange strain
- stress: Stress

### 1.7.10 Subroutine regarding User Defined Materials (**umat.f**)

The interface of the deformation analysis of general materials is provided irrespective of elastic, hyperelastic and elastoplastic materials.

#### 1.7.10.1 (1) Calculation subroutine of stiffness matrix

```
subroutine uMatlMatrix( mname, matl, ftn , stress , fstat , D, temperature , dtime )
  CHARACTER(len=*) , INTENT(IN) :: mname
  REAL(KIND=kreal), INTENT(IN) :: matl(:)
  REAL(KIND=kreal), INTENT(IN) :: ftn(3,3)
  REAL(KIND=kreal), INTENT(IN) :: stress(6)
  REAL(KIND=kreal), INTENT(IN) :: fstat(:)
  REAL(KIND=kreal), INTENT(OUT) :: D(:, :)
  REAL(KIND=kreal), optional :: temperature
  REAL(KIND=kreal), optional :: dtime
```

- mname: Material name
- matl: Array to save the material constants (100 max)
- ftn: Deformation gradient tensor
- stress: 2nd Piola-Kirchhoff stress
- fstat: State variable
- D: Constitutive equation
- temperature: Temperature
- dtime: Time increment

#### 1.7.10.2 (2) Update calculation subroutine of strain and stress

```
subroutine uUpdate( mname, matl, ftn , strain , stress , fstat , temperature , dtime )
  character(len=*) , intent(in) :: mname
  real(KIND=kreal), intent(in) :: matl
  real(kind=kreal), intent(in) :: ftn(3,3)
  real(kind=kreal), intent(inout) :: strain(6)
  real(kind=kreal), intent(inout) :: stress(6)
  real(kind=kreal), intent(inout) :: fstat(:)
  real(KIND=kreal), optional :: temperature
  real(KIND=kreal), optional :: dtime
```

- mname: Material name
- matl: Array to save the material constants (100 max)
- ftn: Deformation gradient tensor
- strain: Strain
- stress: 2nd Piola-Kirchhoff stress
- fstat: State variable
- temperature: Temperature
- dtime: Time increment

### 1.7.11 Process Subroutine of User Defined External Load (`uload.f`)

An interface is provided to process the external load defined by the user.

In order to use the external load defined by the user, first, numerical structure `tULoad` is defined in order to define the external load, and the definition is read using the `!ULOAD` of the input file. Subsequently, the external load is incorporated using the following interfaces.

#### 1.7.11.1 (1) Subroutine for reading external load

```
integer function ureadload( fname )
  character(len=*) , intent(in) :: fname
```

- `fname`: External file name. The user defined external load is read from this file.

#### 1.7.11.2 (2) Subroutine for incorporating the external load into the overall load vector

```
subroutine uloading( cstep , factor , exForce )
  integer , INTENT(IN) :: cstep
  REAL(KIND=kreal) , INTENT(IN) :: factor
  REAL(KIND=kreal) , INTENT(INOUT) :: exForce(:)
```

- `cstep`: Current number of analysis steps
- `factor`: Load factor of current step
- `exForce`: Overall load vector

#### 1.7.11.3 (3) Calculation subroutine of residual stress

```
subroutine uResidual( cstep , factor , residual )
  integer , INTENT(IN) :: cstep
  REAL(KIND=kreal) , INTENT(IN) :: factor
  REAL(KIND=kreal) , INTENT(INOUT) :: residual(:)
```

- `cstep`: Current number of analysis steps
- `factor`: Load factor of current step
- `residual`: Overall residual stress vector

## 1.8 hecmw\_part1

`hecmw_part1` is a utility software which divides a single domain mesh data into sub-domains and generates distributed mesh data for the parallel finite element method.

The software uses the HEC-MW library's data input function to divide single-domain mesh data into an arbitrary number of sub-domains. The data is divided into two regions. In addition, HEC-MW library's data output function is used to generate "distributed mesh data", and the software developed with HEC-MW can analyze the data generated by `hecmw_part1`.

The `hecmw_part1` has two types of domain decomposition methods: "node-based decomposition" and "element-based decomposition".

We have also implemented an interface to the METIS library, which applies graph theory to domain decomposition, so that the domain decomposition method using kMETIS and pMETIS in the library can be implemented. The system also enables segmentation. In addition, it is possible to set the overlapping depth between partial regions in the node-based segmentation.

### 1.8.1 Domain segmentation technique

The following division methods are available for `hecmw_part1`.

#### 1.8.1.1 RCB

It stands for Recursive Coordinate Bisection and is a method of dividing a region based on the size of the coordinate values. It is a fast and stable method, but its subregion is limited to  $2^n$ . It is an effective method for simple shapes.

### 1.8.1.2 METIS

This open-sourced free software is widely used in the world because it is a fast and stable method and provides good region separation even for complex shapes. This software has an interface to the METIS library and allows you to perform region segmentation directly using pMETIS and kMETIS. You can download METIS from the following URL.

METIS - Serial Graph Partitioning and Fill-reducing Matrix Ordering

### 1.8.2 Domain division type

hecmw\_part1 supports the following two types of division of a single region mesh into subregions.

#### 1.8.2.1 Nodal segmentation (Node-based segmentation)

As shown in Figure 1.3-1, this is a node-based region partitioning method, in which the only belonging subregion is determined for all nodes, and overlapping elements between adjacent subregions are created. Therefore, in node-based decomposition, each subregion possesses the following information about nodes and elements (Figure 1.3-2).

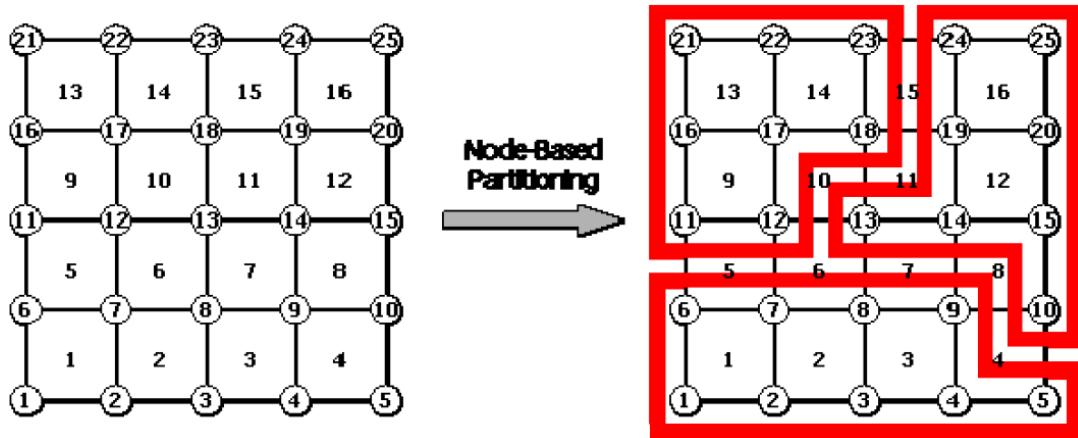


Figure 35: image-20201030104110865

Fig. 1.3-1 Region segmentation per node

Figure 1.3-2 Nodes and elements held by each subregion (node-based segmentation)

- Nodes that belong to that subregion (internal nodes)
- Elements containing internal nodes
- Nodes that make up the elements, including internal nodes

It also possesses the following information on communication between adjacent sub-regions (communication table).

- Import nodes: Nodes within a subregion that belong to another subregion
- Export node: Internal node that is an import node of another sub-region
- Shared elements: elements that are shared with other subdomains

The import nodes, export nodes, and shared elements in the second subregion of Figure 1.3-1 are shown in Figures 1.3-3, 1.3-4, and 1.3-5.

Figure 1.3-3 Import nodes in the second sub-region.

Export nodes in the second sub-region.

Figure 1.3-5 Shared elements in the second subdomain.

#### 1.8.2.2 Region division by elements (element-based division)

As shown in Fig. 1.3-6, region decomposition is performed on an element-by-element basis, and in this case, only one subregion is determined for every element, and overlapping nodes are generated between adjacent subregions. Hence, in element-based decomposition, each subregion has the following nodal points and element information (Figure 1.3-7).

Fig. 1.3-6 Region segmentation by element

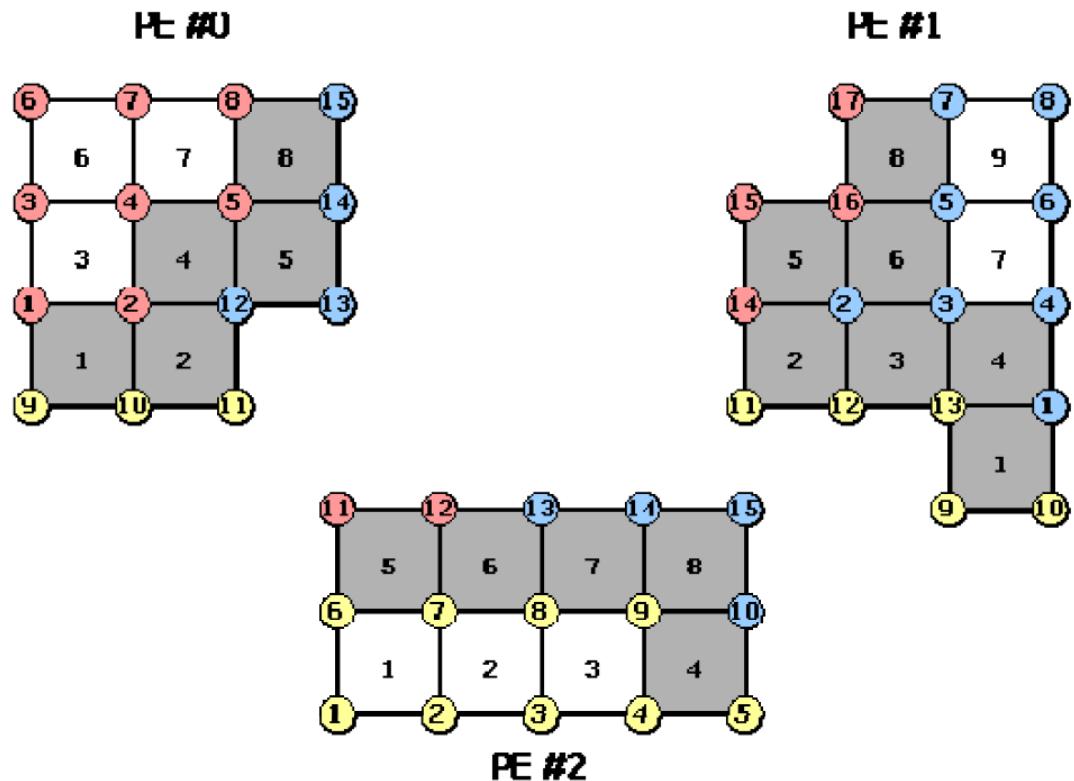


Figure 36: image-20201030104339847

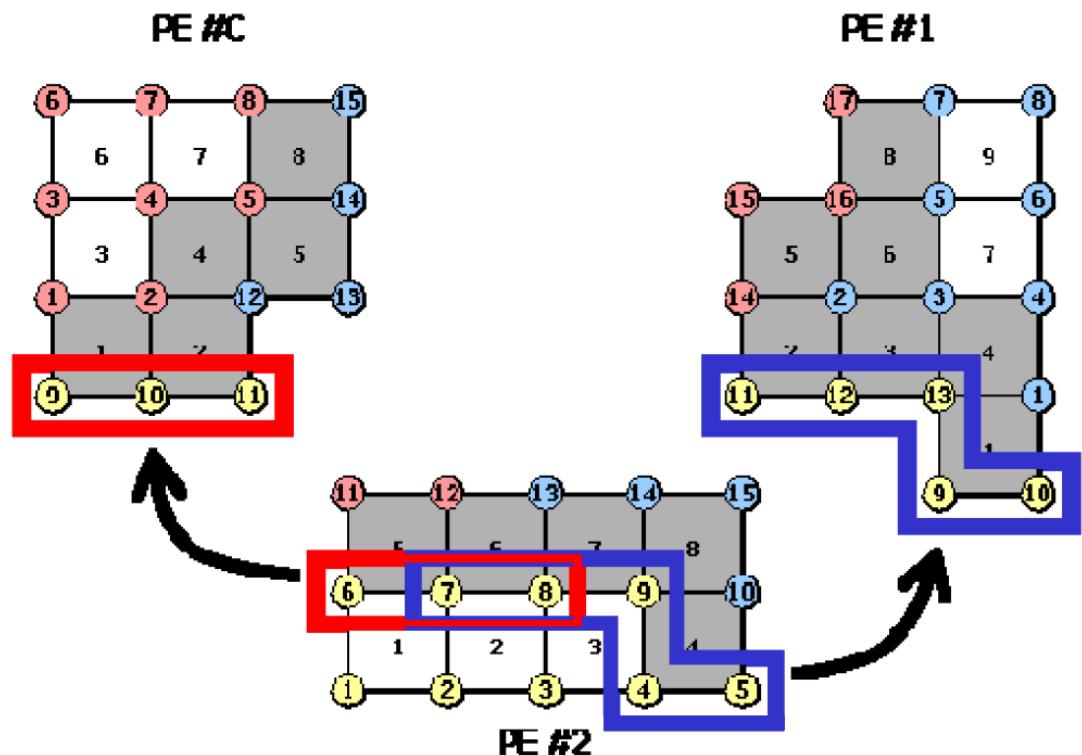


Figure 37: image-20201030104509833

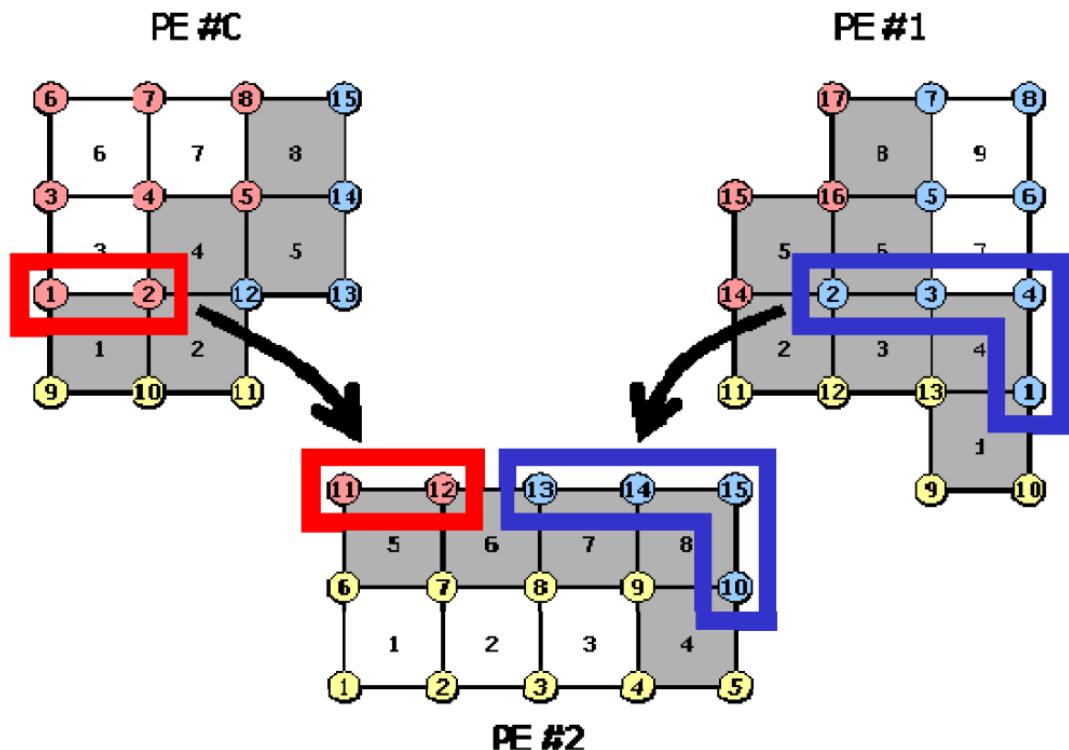


Figure 38: image-20201030104627609

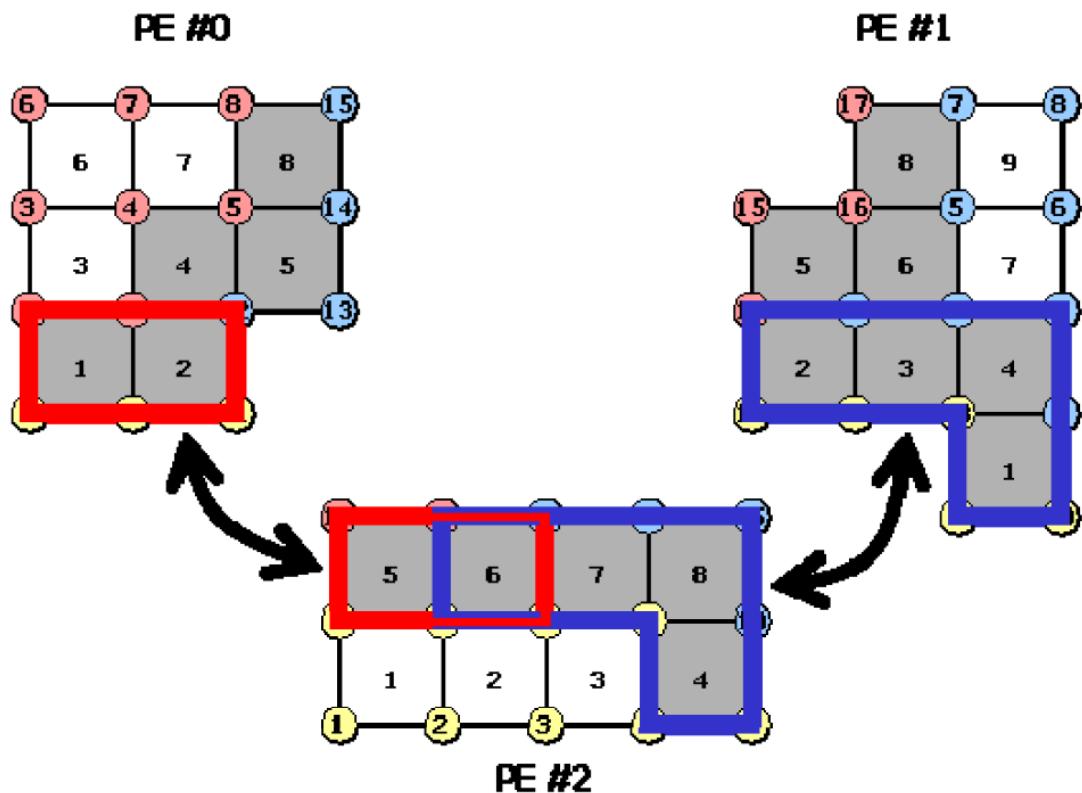


Figure 39: image-20201030104739009

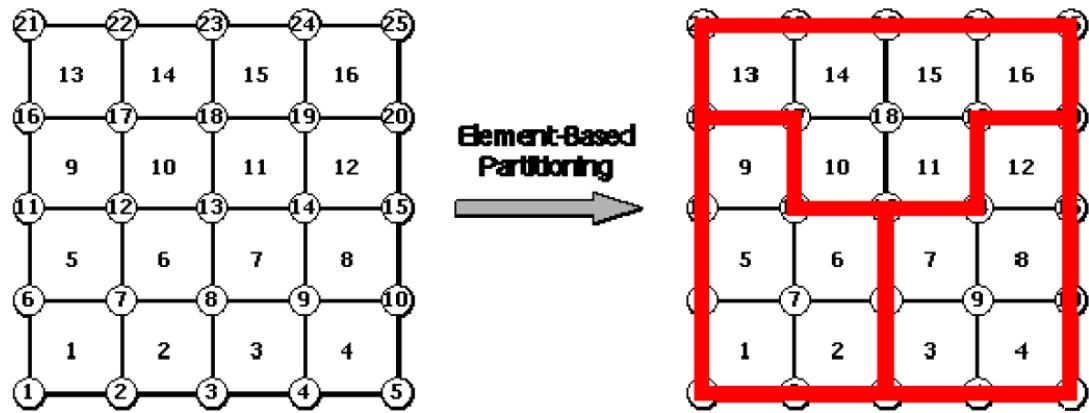


Figure 40: image-20201030104838092

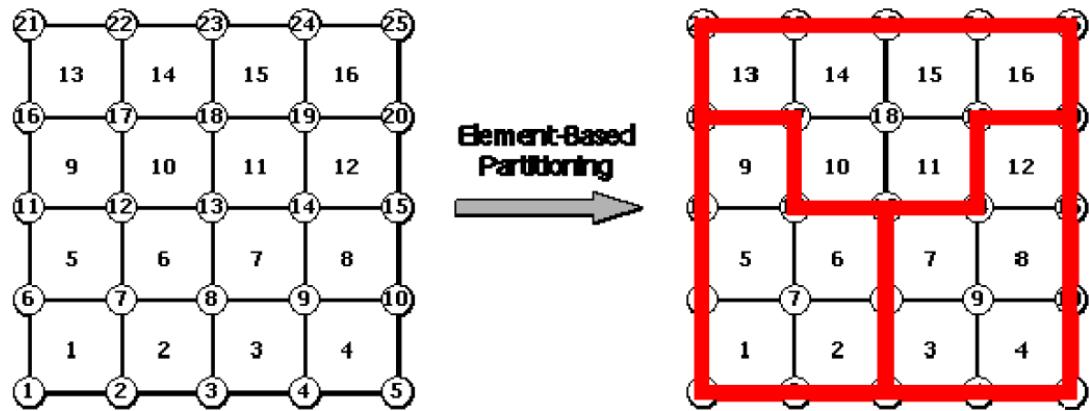


Figure 41: image-20201030104934764

Figure 1.3-7 Nodes and elements held by each subregion (element-based partitioning)

- Elements that belong to this subdomain (internal elements)
- Nodes that make up the internal elements
- Elements containing the nodes that make up the internal elements

It also possesses the following information on communication between adjacent sub-regions (communication table).

- Imported elements: elements in a subregion that belong to another subregion
- Export elements: internal elements that are import elements of other sub-regions
- Shared nodes: nodes that are shared with other subregions

The import element, export element, and shared nodes in the second subregion of Figure 1.3-7 are shown in Figure 1.3-8, Figure 1.3-9, and Figure 1.3-10.

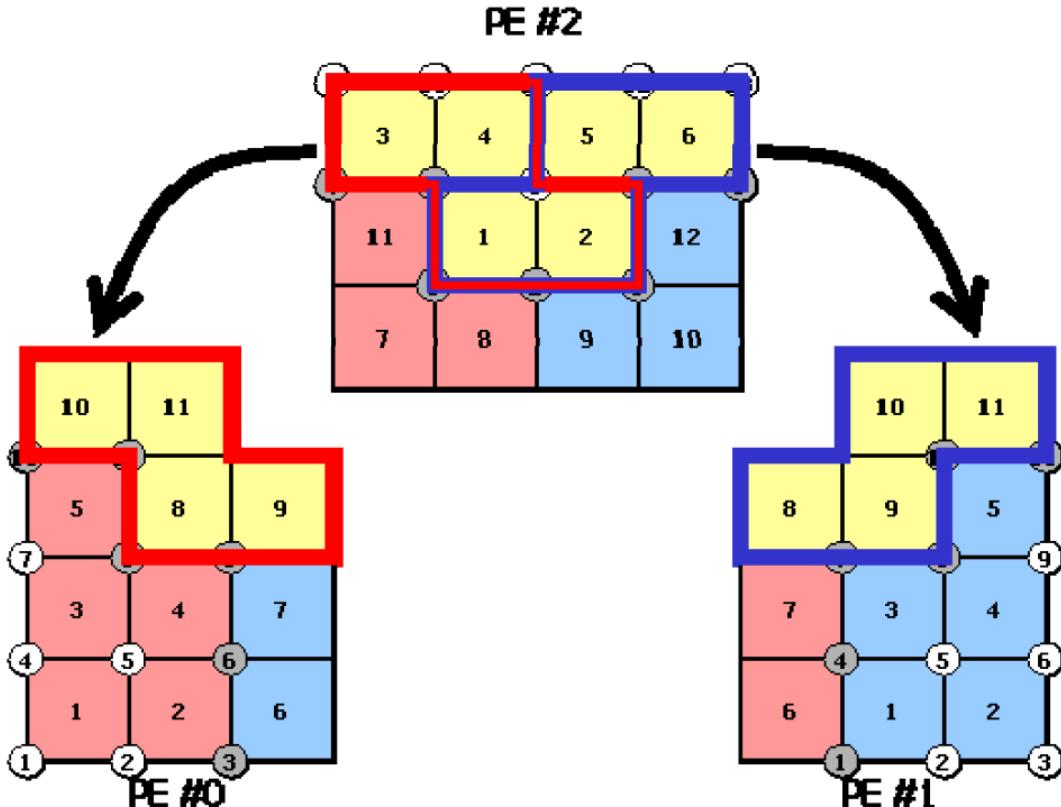


Figure 42: image-20201030105047789

Figure 1.3-8 Import elements in the second sub-region.

Figure 1.3-9 Export elements in the second sub-region.

Fig. 1.3-10 Shared nodes in the second sub-region.

In both types of domain division, the user does not need to consider the communication because the domain division utility automatically creates and writes its communication table to the distributed mesh data.

### 1.8.2.3 Variable overlap depth

The hecmw\_part1 allows you to set the overlapping depth between subregions as you like (except when the region decomposition type is nodal decomposition). Normally, the overlap depth is set to 1. However, when SAI (Sparse Approximate Inverse) is used for preprocessing to solve contact and MPC constraint problems, the overlap depth needs to be increased.

### 1.8.2.4 Creating a UCD file for generating a domain division image

hecmw\_part1 can generate a UCD file to display a region image in MicroAVS and so on.

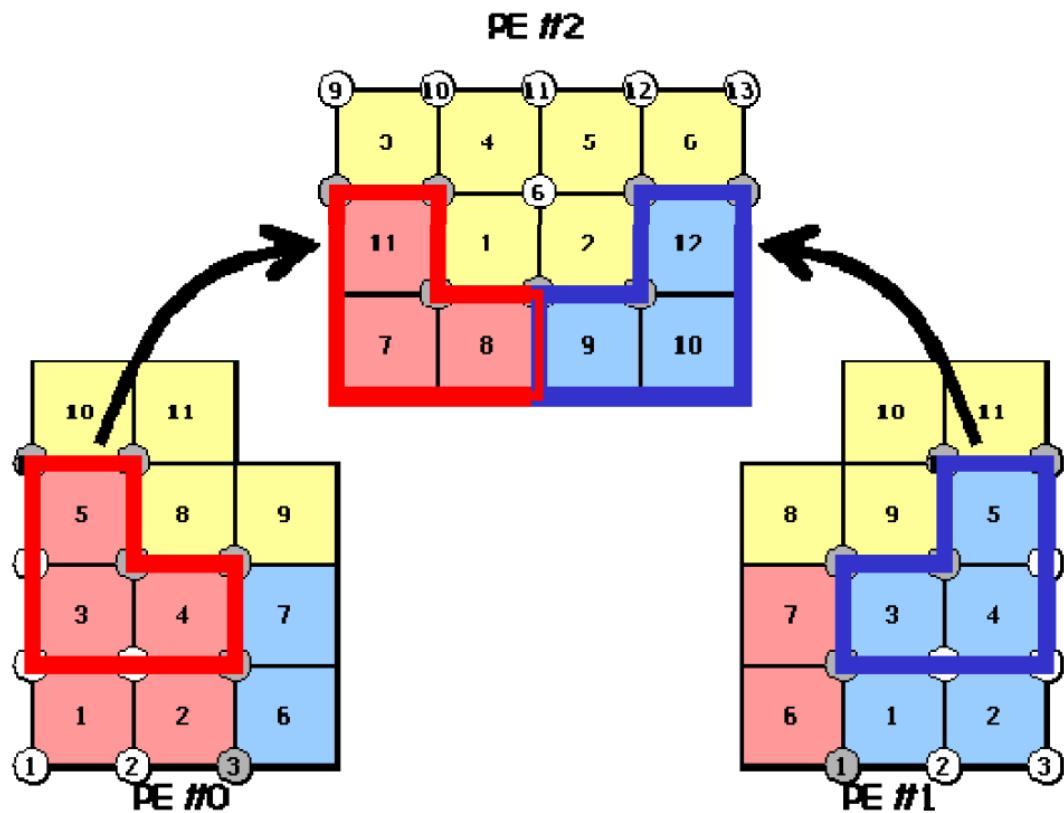


Figure 43: image-20201030105147038

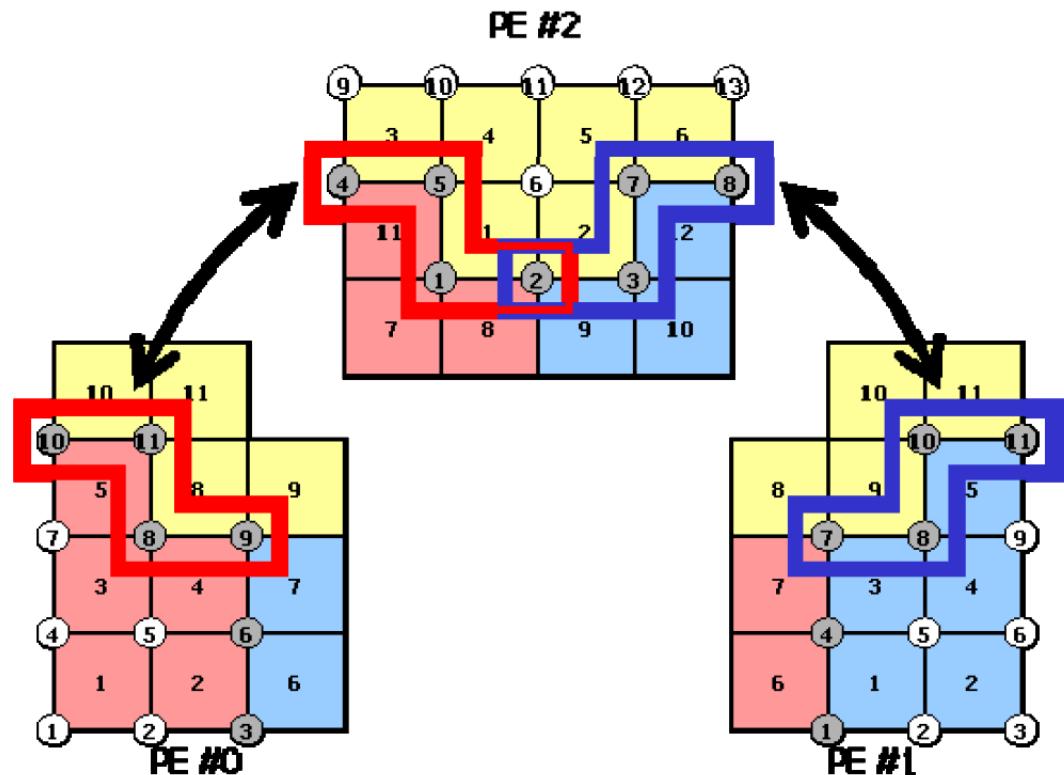


Figure 44: image-20201030105255224

### 1.8.2.5 Handling of contact points

hecmw\_part1 can specify

CONTACT=[DEFAULT|AGGREGATE|DISTRIBUTE|SIMPLE]

for the contact point when dealing with a contact problem.

### 1.8.3 Files required for execution

In order to execute hecmw\_part1, the following files are required.

- Single-region mesh data for domain segmentation
- global control file
- Domain division utility control file

These files are described below.

#### 1.8.3.1 Single-region mesh data file for domain segmentation

The data input function of the HEC-MW library is used for the data input function of this software, and the mesh data of a single region that can be input by this function can be divided into regions as input data.

#### 1.8.3.2 global control file

This software uses the global control file of HEC-MW to specify the mesh data to be displayed. In order to run this software, you need to specify the following in the global control file.

- File name of the mesh data of the single area to be divided
- The file name header for the distributed mesh data to be created

The following is an example of the description of a global control file.

If the single region mesh data named mesh.dat is divided into the following three files, the global control file (hecmw\_ctrl.dat) is written as follows: mesh.dist.0, mesh.dist.1... The description of the global control file (hecmw\_ctrl.dat) is as follows.

```
# Definition of single region mesh data to be domain segmented
!MESH, NAME=part_in, TYPE=HECMW-ENTIRE
mesh.dat
# Define the Distributed Mesh Data to Create
!MESH, NAME=part_out, TYPE=HECMW-DIST
mesh.dist
```

Here, in the definition of the single region mesh data (the first !MESH in the first example), the string specified for the NAME option is fixed at part\_in. In addition, in the definition of the distributed mesh data (the second !MESH in example 2), the string for the NAME option is fixed to part\_out and the file format specifier for the TYPE option is fixed to HECMW-DIST. If these options do not contain the specified string, an error occurs.

#### 1.8.3.3 Domain division utility control file

hecmw\_part1 sets up the conditions of region separation such as the region separation method, the number of region divisions, and so on, in the region separation utility control file (hecmw\_part\_ctrl.dat). The following is how to write the control manual.

The default file name of this control file is hecmw\_part\_ctrl.dat.

In the Application Examples chapter, the contents of the utility control file for each example are shown. Please refer to them when you are writing.

##### 1.8.3.3.1 line 1

```
!PARTITION, TYPE=<type>, METHOD=<method>, DOMAIN=<domain> [ , optional parameter ]
```

Parameter	
TYPE	Domain division type (required)
METHOD	Domain decomposition method (required)
DOMAIN	Number of division (required)
DEPTH	Overlap depth (optional)
UCD	Output UCD file for the segmented image (optional)
CONTACT	

Parameter	Parameter value	Contents
TYPE	NODE-BASED	node-based division
	ELEMENT-BASED	element-based division
METHOD	RCB	RCB method, the division reference axis must be specified on line 2
	KMETIS	kMETIS
	PMETIS	pMETIS
DOMAIN		number of divisions
DEPTH		The overlap depth of the partial region (DEPTH=1 if omitted) Cannot be specified when TYPE=ELEMENT-BASED
UCD		UCD file name (can be abbreviated)
CONTACT	DEFAULT	
	AGGREGATE	
	DISTRIBUTE	
	SIMPLE	

### 1.8.3.3.2 Line 2 (required only if METHOD=RCB)

(Line 2) DIR1, DIR2, DIR3, ... (Specify the number of installments)

variable name	attribute	contents
DIRX	C	Division reference axis (specified by lowercase x, y, or z)

### 1.8.3.3 Example of a control file

Example 1

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=8, DEPTH=1, UCD=mesh.inp
x, y, z
```

Example 2

```
!PARTITION, TYPE=ELEMENT-BASED, METHOD=PMETIS, DOMAIN=32
```

## 1.8.4 execution method

The start-up commands for this software are as follows.

```
$ hecmw_part1 [-f <part_ctrl_filename>]
```

With the -f option, you can specify the file name of the utility control file for region partitioning. If this option is omitted, the utility control information is read from a file with the default file name (hecmw\_part\_ctrl.dat) in the execution directory.

Moreover, when hecmw\_part1 is executed, the global control file is placed in the execution directory and the single region mesh data file is placed in the directory specified in the global control file.

## 1.8.5 Application examples

### 1.8.5.1 Simple cube model (48 nodes x 48 nodes x 48 nodes)

- Element type: Hexahedron linear elements
- Number of nodes : 110,592
- Number of elements: 103,823

#### 1.8.5.1.1 An example of a control file description for the domain division utility

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=64, UCD=cube.rcb.inp
x, y, z, x, y, z
```

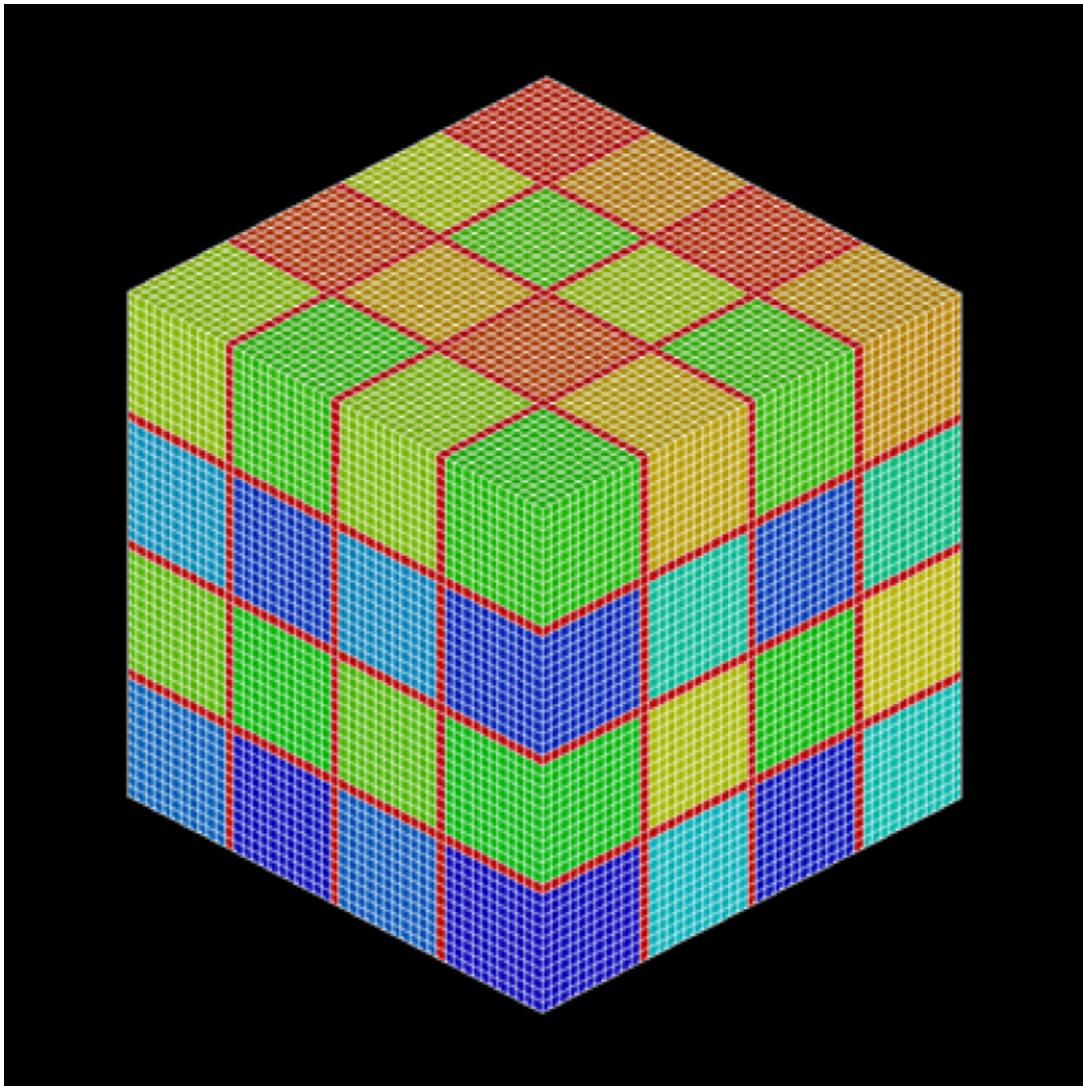


Figure 45: image-20201030105517738

Fig. 4.1-1 Domain decomposition utility application example 1: Simple cube model (Node-based decomposition, RCB method, number of edgecuts: 20,736/324,864)

#### 1.8.5.1.2 An example of a control file description for the domain division utility

```
!PARTITION, TYPE=NODE-BASED, METHOD=KMETIS, DOMAIN=64, UCD=cube.kmetis.inp
```

Fig. 4.1-2 Domain decomposition utility application example 1: Simple cube model (node-based decomposition, kMETIS, edgecut number: 26,160/324,864)

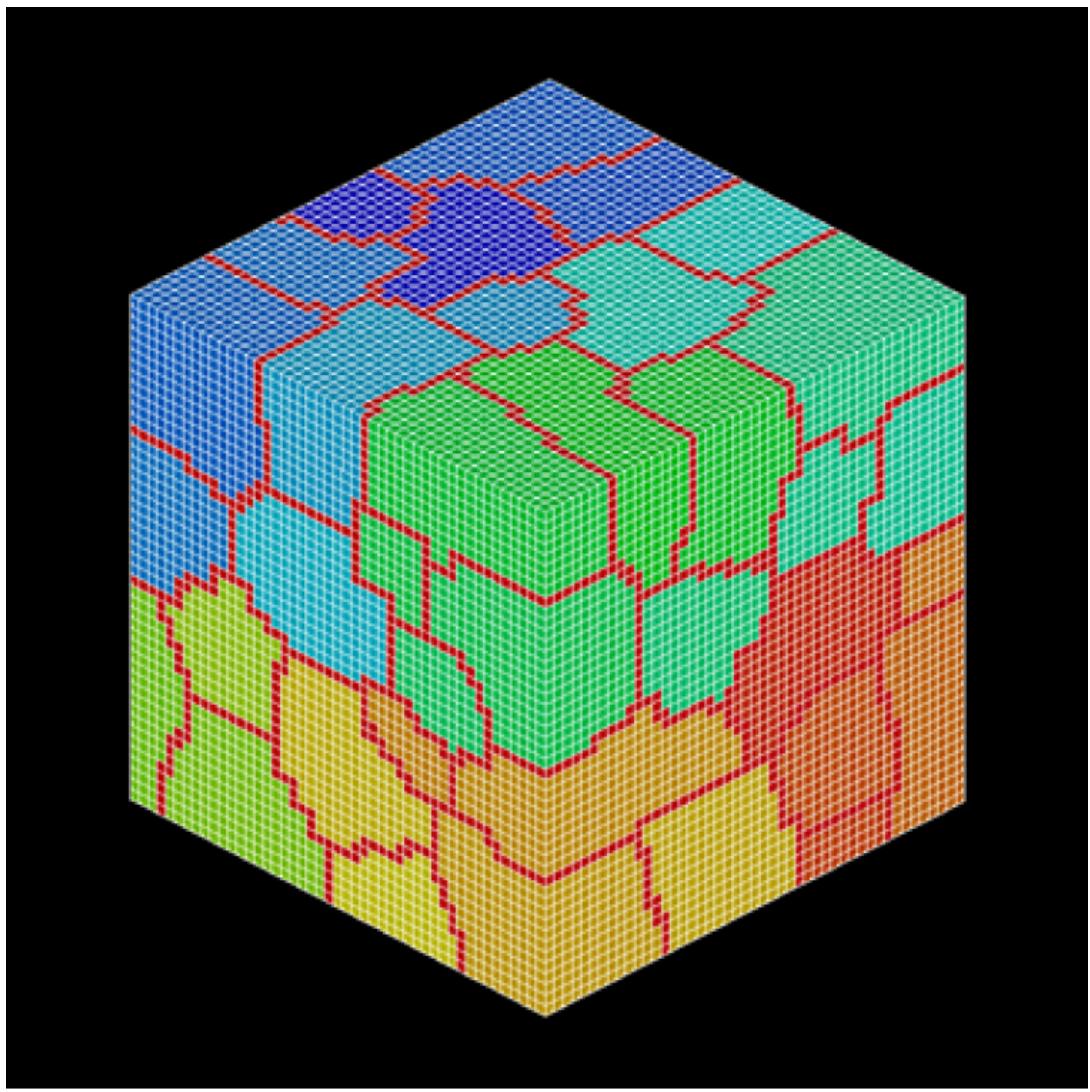


Figure 46: image-20201030105810931

### 1.8.5.2 Akamon (Hongo campus of the University of Tokyo) model

- Element type : Tetrahedral primary elements
- Number of nodes : 3,550
- Number of elements: 10,156

#### 1.8.5.2.1 An example of a control file description for the domain division utility

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=32, UCD=akamon.rcb.inp
x, y, z, x, y
```



Figure 47: image-20201030110040845

Fig. 4.2-1 Domain decomposition utility application example 2: Red gate model (nodal-based decomposition, RCB method, edgecut:4,396/17,180)

#### 1.8.5.2.2 An example of a control file description for the domain division utility

```
!PARTITION, TYPE=NODE-BASED, METHOD=PMETIS, DOMAIN=32, UCD=akamon.kmetis.inp
```

Fig.4.2-2 Domain decomposition utility application example 2: Red gate model (nodal-based decomposition,pMETIS,edgecut:2,119/17,180)

### 1.8.5.3 Honshu model



Figure 48: image-20201030110233242

- Element type: triangular primary element
- Number of nodes : 21,285
- Number of elements: 40,548

#### 1.8.5.3.1 Example of a control file description for the domain division utility

```
!PARTITION, TYPE=ELEMENT-BASED, METHOD=RCB, DOMAIN=64, UCD=honshu.rcb.inp
x, y, x, y, x, y
```

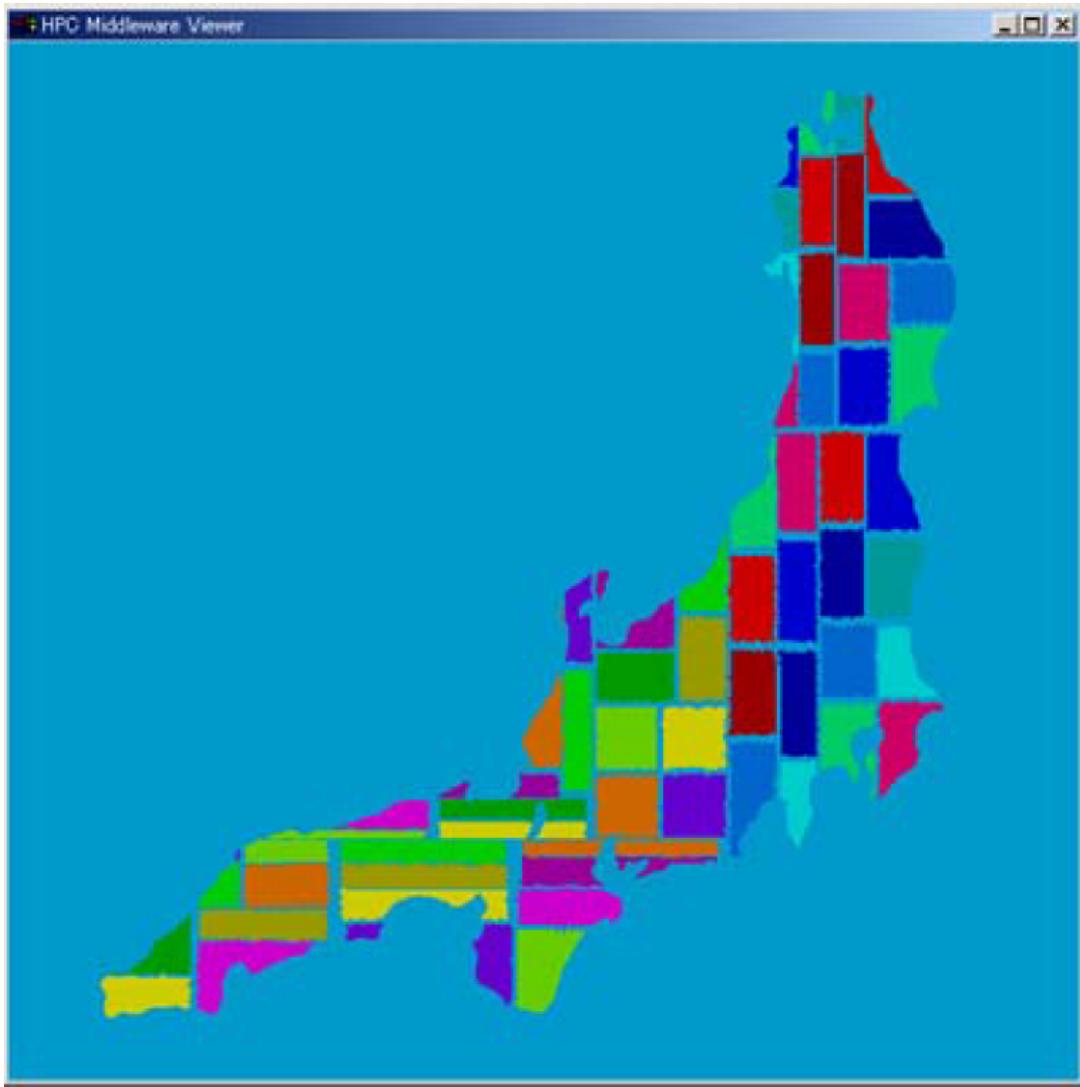


Figure 49: image-20201030110424834

Fig.4.3-1 Utility application example 3: Honshu model (element-based decomposition, RCB method, edge-cut:17,241/237,627)

#### 1.8.5.3.2 Example of a control file description for the domain division utility

```
!PARTITION, TYPE=ELEMENT-BASED, METHOD=KMETIS, DOMAIN=64, UCD=honshu.kmetis.inp
```

Fig.4.3-2 Domain decomposition utility application example 3: Honshu model (element-based decomposition,kMETIS,edgecut:11,657/237,627)

#### 1.8.5.4 Graphite Block Model

- Element type: primary hexahedron elements

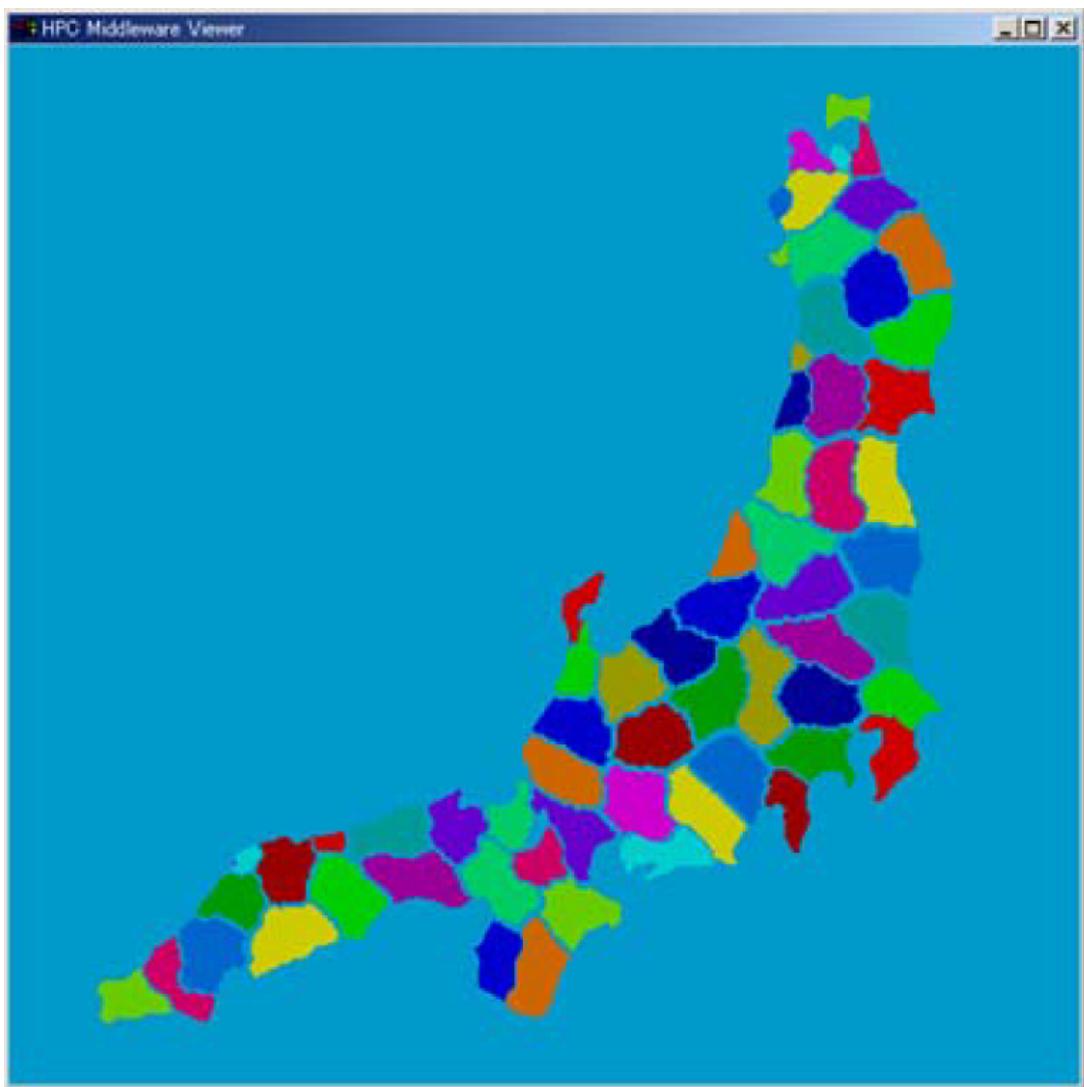


Figure 50: image-20201030110620826

- Number of nodes : 1,308
- Number of elements: 795

#### 1.8.5.4.1 An example of a control file description for the domain division utility

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=4, DEPTH=1, UCD=carbon.1.inp
x, y
```

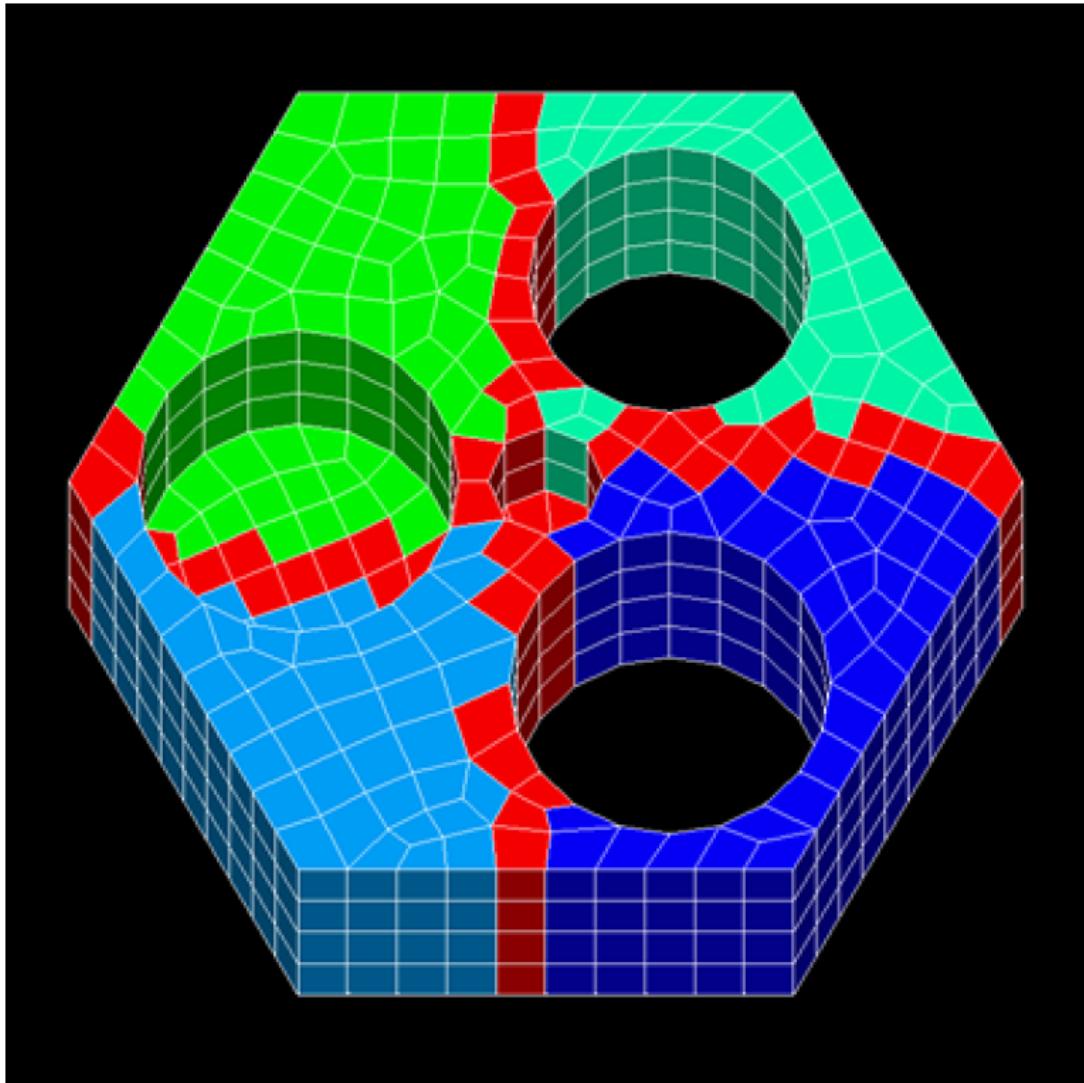


Figure 51: image-20201030110812168

Fig.4.4-1 Application example of domain decomposition utility 4: Graphite block model (overlap depth change example, DEPTH=1)

#### 1.8.5.4.2 An example of a control file description for the domain division utility

```
!PARTITION, TYPE=NODE-BASED, METHOD=RCB, DOMAIN=4, DEPTH=2, UCD=carbon.2.inp
x, y
```

Fig.4.4-2 Application example of domain decomposition utility 4: Graphite block model (overlap depth change example, DEPTH=2)

### 1.8.6 error message

#### 1.8.6.1 HECMW-PART-E0001: No such file or directory

The specified file does not exist.

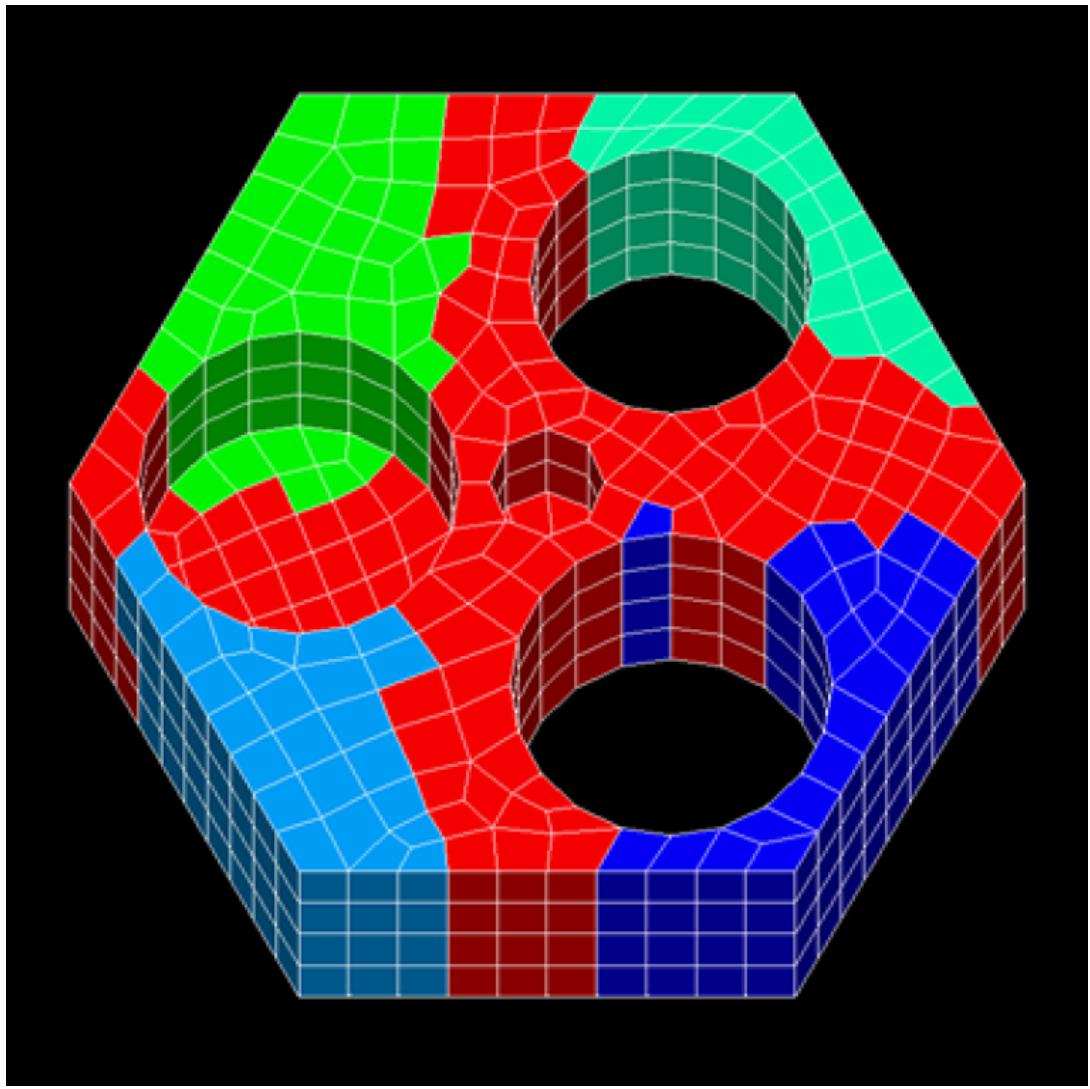


Figure 52: image-20201030111036176

**1.8.6.2 HECMW-PART-E0002: File close error**

An unexpected error occurred while closing the file.

**1.8.6.3 HECMW-PART-E0003: Too long file name**

The file name is too long.

**1.8.6.4 HECMW-PART-E0004: NULL pointer was found**

Illegal null pointer detected.

**1.8.6.5 HECMW-PART-E0005: Invalid EOF is found**

Unexpected EOF detected.

**1.8.6.6 HECMW-PART-E0006: Invalid argument is found**

Illegal arguments are detected.

**1.8.6.7 HECMW-PART-E0101: Invalid token found**

Invalid token detected.

**1.8.6.8 HECMW-PART-E0111: 'TYPE' must not be omitted**

Can't find 'TYPE', a mandatory option for '!PARTITION'

**1.8.6.9 HECMW-PART-E0121: 'METHOD' must not be omitted**

Can't find the required option 'METHOD' for '!PARTITION'

**1.8.6.10 HECMW-PART-E0131: 'DOMAIN' must not be omitted**

Missing 'DOMAIN', a mandatory option for '!PARTITION'

**1.8.6.11 HECMW-PART-E0134: 'METHOD' is 'RCB', but 'DOMAIN' is not n-th power of 2**

'RCB' is specified for 'METHOD', but the value of 'DOMAIN' is not a power of 2.

**1.8.6.12 HECMW-PART-E0161: Partitioning directions are not enough**

The direction of divisional axis is not specified enough.

**1.8.6.13 HECMW-PART-E0201: Invalid partitioning type is found**

We have detected an illegal domain split type.

**1.8.6.14 HECMW-PART-E0202: Invalid partitioning method is found**

We have detected an illegal domain division method.

**1.8.6.15 HECMW-PART-E0203: Invalid element type is found**

Illegal element type detected.

**1.8.6.16 HECMW-PART-E0204: Invalid partitioning direction is found**

Illegal split axis direction detected.

**1.8.6.17 HECMW-PART-E0205: Invalid number of sub-domains is found**

We have detected an illegal subspace count.

**1.8.6.18 HECMW-PART-E0206: Invalid overlapping depth is found**

Illegal inter-regional overlap depth detected

**1.8.6.19 HECMW-PART-E0211: Stack overflow**

The stack array overflowed.

**1.8.6.20 HECMW-PART-E0301: Domain number is outside of the range**

The number of partial regions exceeds the number of partial regions.

**1.8.6.21 HECMW-PART-E0311: Number of nodes is outside of the range**

Illegal node count detected.

**1.8.6.22 HECMW-PART-E0312: Number of internal nodes is outside of the range**

Illegal internal node count detected.

**1.8.6.23 HECMW-PART-E0321: Number of elements is outside of the range**

Illegal element count detected.

**1.8.6.24 HECMW-PART-E0322: Number of internal elements is outside of the range**

We have detected an illegal internal element count.

**1.8.6.25 HECMW-PART-E0331: Number of neighboring sub-domains must be grater than or equal 0**

The number of adjacent subregions is less than 0.

**1.8.6.26 HECMW-PART-E0501: Log file has not initialized yet**

Log files have not been initialized.

**1.8.6.27 HECMW-PART-E9999: Fatal error**

There was a serious error.

**1.8.6.28 HECMW-PART-W0162: Too many partitioning directions are specified**

More than the required number of axis directions are specified.

**1.8.6.29 HECMW-PART-W0401: EQUATION\_BLOCK is not found**

Can't find the EQUATION\_BLOCK.

**1.8.6.30 HECMW-PART-W0502: Log file for practitioner has already initialized**

The log file initialization process was called multiple times.

**1.9 hecmw\_vis1**

**1.10 hec2rcap**

**1.11 rconv**

**1.12 rmerge**