

- [1 FrontISTR Analysis Flow Manual](#)
  - [1.1 Manuals](#)
  - [1.2 List of description on this manual](#)
  - [1.3 Analysis Flow and Input/Output File](#)
    - [1.3.1 Analysis Flow](#)
    - [1.3.2 Overall Control Data](#)
    - [1.3.3 Mesh Data](#)
    - [1.3.4 Analysis Control Data](#)
    - [1.3.5 Output File](#)
    - [1.3.6 Execution Procedure](#)
  - [1.4 Element Library and Material Data](#)
    - [1.4.1 Element Library](#)
    - [1.4.2 Material Data](#)
    - [1.4.3 Nonlinear Static Analysis](#)
  - [1.5 Overall Control Data](#)
    - [1.5.1 Outline of Overall Control Data](#)
    - [1.5.2 Input Rules](#)
    - [1.5.3 Header List](#)
  - [1.6 Single Domain Mesh Data](#)
    - [1.6.1 Outline of Single Mesh Data](#)
    - [1.6.2 Input Rules](#)
    - [1.6.3 Header List of Single Domain Mesh Data](#)
  - [1.7 Analysis Control Data](#)
    - [1.7.1 Outline of Analysis Control Data](#)
    - [1.7.2 Input Rules](#)
    - [1.7.3 Analysis Control Data](#)
  - [1.8 User Subroutines](#)
    - [1.8.1 Input of User Defined Material](#)
    - [1.8.2 Subroutine regarding Elastoplasticity Deformation \(`uyield.f90`\)](#)
    - [1.8.3 Subroutine regarding Elastic Deformation \(`uelastic.f90`\)](#)
    - [1.8.4 Subroutine regarding User Defined Materials \(`umat.f`\)](#)
    - [1.8.5 Process Subroutine of User Defined External Load \(`uload.f`\)](#)
  - [1.9 User Subroutines](#)
    - [1.9.1 Input of User Defined Material](#)
    - [1.9.2 Subroutine regarding Elastoplasticity Deformation \(`uyield.f90`\)](#)
    - [1.9.3 Subroutine regarding Elastic Deformation \(`uelastic.f90`\)](#)
    - [1.9.4 Subroutine regarding User Defined Materials \(`umat.f`\)](#)
    - [1.9.5 Process Subroutine of User Defined External Load \(`uload.f`\)](#)
  - [1.10 ステップ制御](#)
    - [1.10.1 解析上の時間について](#)
    - [1.10.2 静解析の制御](#)
    - [1.10.3 動解析の制御](#)

# 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	5.0

License	MIT License
Correnponding Clerks	FrontISTR Commons 2-11-16 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](#)

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 analyais 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, th eigenvalue analysis and heat conduction analysis method is described.

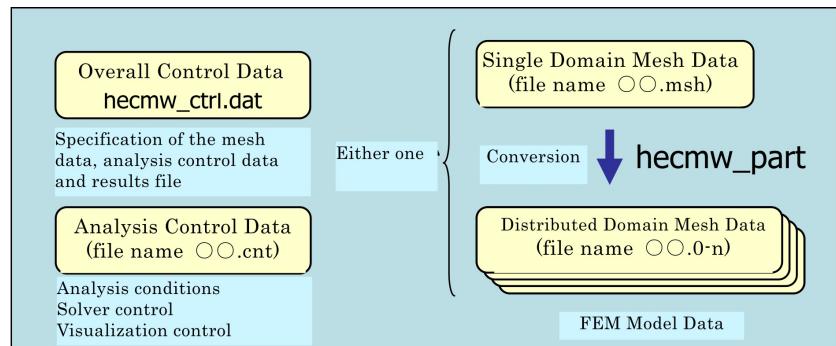
## 1.2 List of description on this manual

- [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](#)

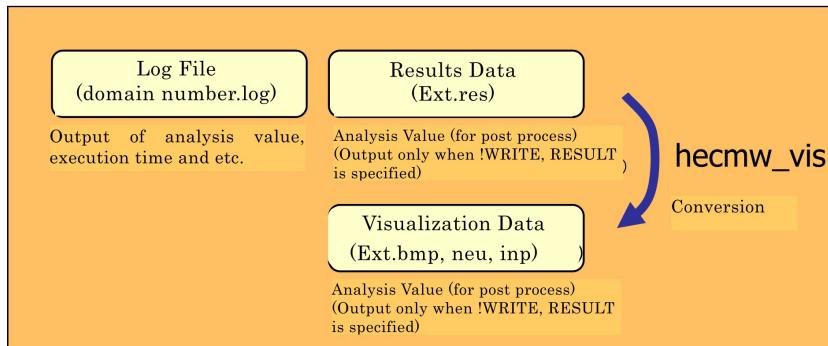
## 1.3 Analysis Flow and Input/Output File

### 1.3.1 Analysis Flow

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



(a) Input File



(b) 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* is 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=fst rRES,I0=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
!MATERIAL,NAME=M1,ITEM=1        ----- Definition of section data
!ITEM=1,SUBITEM=2              ----- Definition of material data
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
conditions
  FIX,1,3,0.0
!CLOAD
  CL1,1,-1.0 ----- Definition of concentrated load conditions
!DLOAD
  ALL,BX,1.0 ----- Definition of distributed load conditions
!REFTEMP
  20.0 ----- Definition of reference temperature
!TEMPERATURE
coditions
  ALL,100.0 ----- Definition of heat load (temperature)
!! 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
visualizer
!surface_num=1
!surface_style=1
!END
----- Hereinafter, the control data of the

```

### 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

FrontISTR requires `hecmw_ctrl.dat`, analysis control data and distributed domain mesh data

for parallel execution in general. Parallel domain contact analysis does not require distributed domain mesh as it partitions the domain after reading the single domain mesh data.

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

The all other processes of execution procedure are the same as the other analysis.

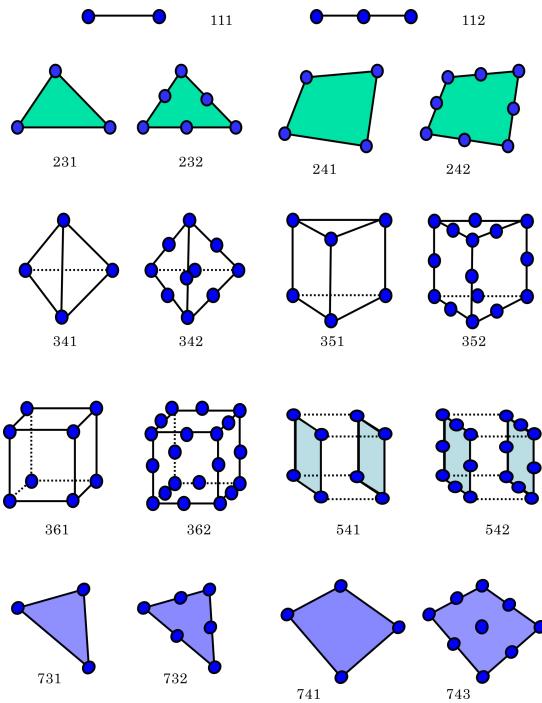
## 1.4 Element Library and Material Data

### 1.4.1 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 element
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)

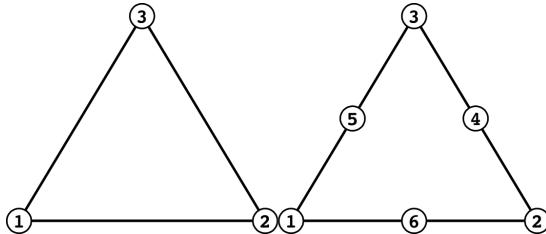


**Figure 4.1.1: Element Library**

#### 1.4.1.1 (Line Element)



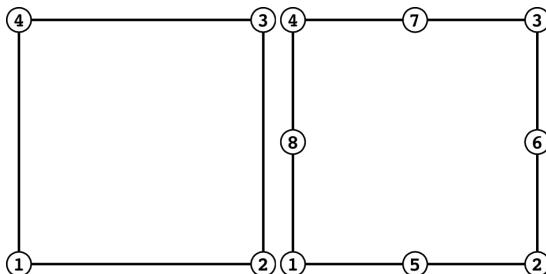
#### 1.4.1.2 (Triangular Plane Element)



##### Surface No. Linear Quadratic

1	1 - 2	1 - 6 - 2
2	2 - 3	2 - 4 - 3
3	3 - 1	3 - 5 - 1

#### 1.4.1.3 (Quadrilateral Plane Element)

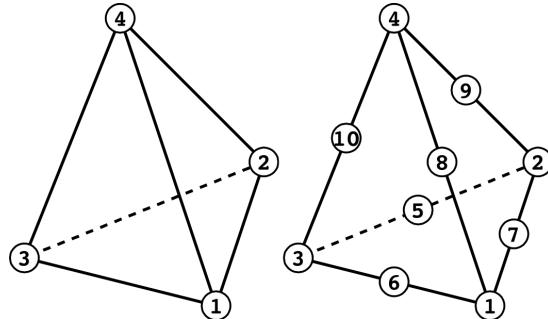


##### Surface No. Linear Quadratic

1	1 - 2	1 - 5 - 2
---	-------	-----------

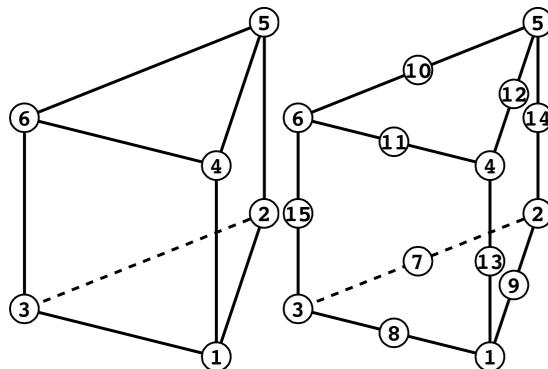
2	2 - 3	2 - 6 - 3
3	3 - 4	3 - 7 - 4
4	4 - 1	4 - 8 - 1

#### 1.4.1.4 (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

#### 1.4.1.5 (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

#### 1.4.1.6 (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

#### 1.4.1.7 (Beam Element)

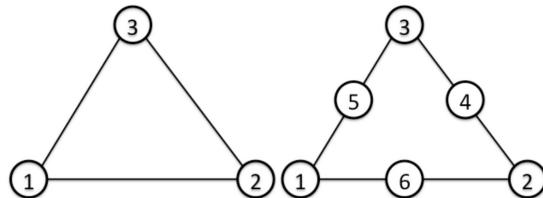


#### 1.4.1.8 (Beam Element with 3-dof nodes)



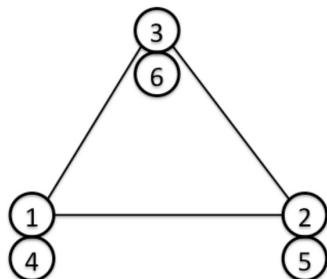
Nodes 1 and 2 for translational DOF, nodes 3 and 4 for rotational DOF.

#### 1.4.1.9 (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]

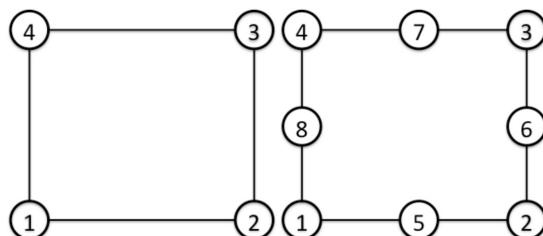
#### 1.4.1.10 (Triangular Shell Element with 3-dof nodes)



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

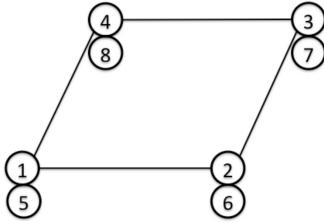
Surface No.	Linear
1	1 - 2 - 3 [front]
2	3 - 2 - 1 [back]

#### 1.4.1.11 (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]

#### 1.4.1.12 (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**

## 1.4.2 Material Data

### 1.4.2.1 Elastic Static Analysis, Linear Dynamic Analysis and Eigenvalue Analysis

In the elastic static analysis and eigenvalue analysis of FrontISTR, it is necessary to use isotropic elastic material 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.

### 1.4.2.2 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.

#### **1.4.2.2.1 (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.

#### **1.4.2.2.2 (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

#### **1.4.2.2.3 (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  $\theta$

Orthotropy angle is specified in  $\theta$  [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  $\theta$  (1st layer)
- 10th parameter : Young's modulus E1 (2nd layer)
- 11th parameter : Poisson's ratio  $n_{12}$  (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  $\theta$  (2nd layer)

### 1.4.3 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
```

## 1.5 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**

file

**Contents**

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**

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 内容**

NAME	Identifier (mandatory)	
IO	Input/output specification (mandatory)	
<b>Parameter Name Parameter Value Contents</b>		
NAME	<name>	Identifier
IO	IN	For input
	OUT	For output
	INOUT	Common to input/output
<b>Parameter Name</b>		<b>Contents</b>
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)	

		<b>Parameter Name Parameter Value</b>	<b>Contents</b>
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

fileheader		<b>Contents</b>
		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 Us
```

```
!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 subdirectories 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 "!", 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

!CONTROL	Specify of analysis control file
!MESH	Specify of mesh file
!RESTART	Specify of restart file
!RESULT	Specify of analysis results file
!SUBDIR	Specify of making sub-directories

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
!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)

<b>Parameter Name</b>	<b>Parameter Value</b>	<b>Contents</b>
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 used together with the 2nd line or later

<b>Parameter Name Attributions</b>	<b>Contents</b>
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)

<b>Parameter</b>	<b>Parameter</b>	<b>Contents</b>
------------------	------------------	-----------------

Name	Value	
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 used 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
"501, 505" are added to group "EA01".
!EGROUP, EGRP=EA04, GENERATE
added to
301, 309, 2
"301, 303, 305, 307, 309, 311, 312, 313" are
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 Name	Parameter Value	Contents
TYPE	111 231 232 241 242 301 341 342 351 352 361 362 541 611 641 731 741 743 761 781	Rod, link element (Linear) Triangular element (Linear) Triangular element (Quadratic) Quadrilateral element (Linear) Quadrilateral element (Quadratic) Truss element (Linear) Tetrahedral element (Linear) Tetrahedral element (Quadratic) Triangular prism element (Linear) Triangular prism element (Quadratic) Hexahedral element (Linear) Hexahedral element (Quadratic) Interface element (Quadrilateral cross section, Linear)
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)

<b>Parameter Name</b>	<b>Parameter Value</b>	<b>Contents</b>
<b>INPUT</b>	<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)

<b>Parameter Name Attributions</b>		<b>Contents</b>
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	Parameter	Contents
Name	Value	
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.)

<b>Subparameter Name</b>	<b>Parameter Value</b>	<b>Contents</b>
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
```

<b>Parameter Name Attributions</b>		<b>Contents</b>
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
```

<b>Parameter Name Attributions</b>		<b>Contents</b>
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
  35.0
----- No temperature dependency
```

```

!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 Number of physical property items by user definition (1 or more)
ITEM	<ITEMnum>	<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
8.0102E-10

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

```

#### 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>

```

<b>Parameter Name</b>	<b>Parameter Value</b>	<b>Contents</b>
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

INPUT	<filename>	nodes belonging to the node group 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, 303, 305, 307, 309, 311, 312, 313
----- "301, 303, 305, 307, 309, 311, 312, 313"
311, 313
----- are added to group "NA04".
```

### 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		Parameter	Contents
Name	Value		
SYSTEM	R	Cartesian coordinate system (Default value)	
	C	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** Parameter

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

<b>Parameter Name Attributions</b>		<b>Contents</b>
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

<b>Parameter Name Attributions</b>		<b>Contents</b>
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

<b>Parameter Name Attributions</b>		<b>Contents</b>
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

<b>Parameter Name Attributions</b>		<b>Contents</b>
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 Parameter**

Name	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)
```

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
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
602, 2
"(601,1) and (602 2)" are added to group "SUF01".
```

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
!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

<b>Parameter Name Attributions</b>	<b>Contents</b>
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 (ommissible)

<b>Parameter Name</b>	<b>Parameter Value</b>	<b>Contents</b>
TYPE	NODE-SURF	Slave surface is the node group Master surface is the surface group (default)
	SURF-SURF	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

### **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

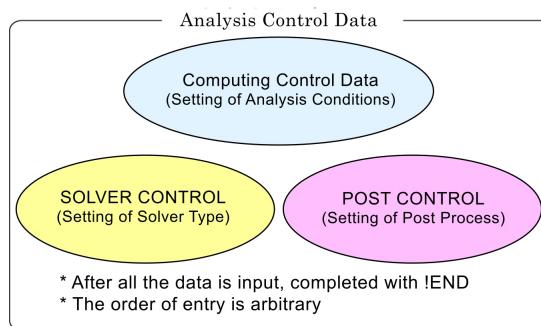
#### Parameter

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.



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=2,ITERLOG=NO,TIMELOG=NO
100,
2
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 7.3.2: Control Data for Static Analysis**

Header	Meaning	Remarks No.	Description
!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	
!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	

<b>!TEMPERATURE</b>	Nodal temperature in thermal stress analysis	2-9
<b>!REFTEMP</b>	Reference temperature in thermal stress analysis	2-10
<b>!STEP</b>	Analysis step control	2-11

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

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

**Parameter Name Attributions                          Contents**

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

**1.7.3.1.0.0.1** 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

**1.7.3.1.1 (13) !END (1-13)**

Displays the end of the control data.

**1.7.3.1.1.1 Parameter**

N/A

**1.7.3.2 Control Data for Static Analysis**

**1.7.3.2.1 (1) !STATIC (2-1)**

Performs the static analysis. (Default: ommissible)

**1.7.3.2.1.1 Parameter**

N/A

**1.7.3.2.2 (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.

#### 1.7.3.2.2.1 Parameter

NAME = Material name

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

Definition of elastic material

#### 1.7.3.2.3.1 Parameter

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

#### 1.7.3.2.3.2 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, ν12, ν13, ν23, G12, G13, G23, Temperature

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

- In the case of TYPE=USER

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

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

Definition of plastic material

#### 1.7.3.2.4.1 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 \*\*

#### 1.7.3.2.4.1.1 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) ε₀, K, n

** In case of HARDEN = RAMBERG-OSGOOD **

(2nd line) ε₀, D, n

** In case of HARDEN = KINEMATIC **

(2nd line) YIELD0, C

** In case of HARDEN = COMBINED **

(2nd line) YIELD0, H, C

1.7.3.2.4.1.2 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	Attributions	Contents
YIELD0	R	Initial yield stress
H	R	Hardening factor
PSTRAIN	R	Plastic strain
YIELD	R	Yield stress
ε₀, K, n	R	$\bar{\sigma} = k(\varepsilon_0 + \bar{\varepsilon})^n$
ε₀, D, n	R	$\varepsilon = \frac{\sigma}{E} + \varepsilon_0 \left(\frac{\sigma}{D}\right)^n$
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

#### 1.7.3.2.4.2 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.

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

Definition of hyperelastic material

##### **1.7.3.2.5.1 Parameter**

```

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

```

##### **1.7.3.2.5.2 2nd Line or later**

###### **1.7.3.2.5.2.1 TYPE = NEOHOOKEの場合**

(2nd line) C<sub>10</sub>, D

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
C <sub>10</sub>	R	Material constant
D	R	Material constant

###### **1.7.3.2.5.2.2 In case of TYPE = MOONEY-RIVLIN**

(2nd line) C<sub>10</sub>, C<sub>01</sub>, D

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
-----------------------	---------------------	-----------------

$C_{10}$	R	Material constant
$C_{01}$	R	Material constant
D	R	Material constant

1.7.3.2.5.2.3 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

1.7.3.2.5.2.4 In case of TYPE = USER

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

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

Definition of viscoelastic material

##### 1.7.3.2.6.1 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

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

Definition of creep material

##### 1.7.3.2.7.1 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要)

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

Definition of mass density

##### 1.7.3.2.8.1 Parameter

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

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

(2nd line) density

<b>Parameter Name Attributions</b>		<b>Contents</b>
density	R	Mass density

#### **1.7.3.2.9 (9) !EXPANSION\_COEFF (2-2-7)**

Definition of coefficient of linear expansion

##### **1.7.3.2.9.1 Parameter**

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

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

##### 1.7.3.2.9.1.1 In case of TYPE=ISOTROPIC

(2nd line) expansion, Temperature

##### 1.7.3.2.9.1.2 In case of TYPE=ORTHOTROPIC

(2nd line) α11, α22, α33, Temperature

<b>Parameter Name Attributions</b>		<b>Contents</b>
expansion	R	Coefficient of thermo expansion
α11, α22, α33	R	Coefficient of thermo expansion
Tempearture	R	Temperature (required when DEPENDENCIES = 1)

#### **1.7.3.2.10 (10) !TRS (2-2-8)**

Thermorheological Simplicity description on temperature behavior of viscoelastic materials

##### **1.7.3.2.10.1 Parameter**

```
DEFINITION = WLF(Default) /ARRHENUS
```

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

(2nd line) θ₀, C₁, C₂

<b>Parameter Name Attributions</b>		<b>Contents</b>
θ₀	R	Reference temperature
C₁, C₂	R	Material constants

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

Definition of flow condition

##### **1.7.3.2.11.1 Parameter**

```
TYPE = INCOMP_NEUTONIAN (Default)
```

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

(2nd line) μ

<b>Parameter Name Attributions</b>		<b>Contents</b>
μ	R	Viscosity

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

Input of user defined material

#### **1.7.3.2.12.1 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

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

Definition of displacement boundary conditions

##### **1.7.3.2.13.1 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

<b>Parameter Name Attributions</b>		<b>Contents</b>
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)

##### **1.7.3.2.13.2 Example of Use**

```
!BOUNDARY, GRPID=1
  1, 1, 3, 0.0
  ALL, 3, 3,
```

Note: Restricted value is 0.0

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

Definition of spring boundary conditions

##### **1.7.3.2.14.1 Parameter**

GRPID = Group ID

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

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

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

##### **1.7.3.2.14.2 Example of Use**

```
!SPRING, GRPID=1
  1, 1, 0.5
```

#### **1.7.3.2.15 (15) !CLOAD (2-4)**

Definition of concentrated load

##### **1.7.3.2.15.1 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

#### 1.7.3.2.15.2 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

```

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

Definition of distributed load

##### 1.7.3.2.16.1 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)

##### 1.7.3.2.16.1.1 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
	Applies pressure to		

P2	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

#### 1.7.3.2.16.1.2 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, 0.0, 1.0
```

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

Input of user definition load

##### 1.7.3.2.17.1 Parameter

FILE = file name (Mandatory)

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

Specification of the contact analysis algorithm

##### 1.7.3.2.18.1 Parameter

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

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

Definition of contact conditions

##### 1.7.3.2.19.1 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

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

#### 1.7.3.2.19.2 Example of Use

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

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

Specification of nodal temperature used for thermal stress analysis

##### 1.7.3.2.20.1 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)

** 2nd line or later **

(2nd line) NODE_ID, Temp_Value
```

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

#### 1.7.3.2.20.2 Example of Use

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

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

Definition of reference temperature in thermal stress analysis

##### 1.7.3.2.21.1 Parameter

N/A

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

(2nd line) Value

<b>Parameter Name Attributions</b>		<b>Contents</b>
Value	R	Reference temperature (Default: 0)

#### 1.7.3.2.22 (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.

#### 1.7.3.2.22.1 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!=!LOAD, !DLOAD, !SPRING, !TEMPERATURE
CONTACT, id          GRPID defined in id!=!CONTACT
```

#### 1.7.3.2.22.2 Parameter

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

### 1.7.3.3 Control Data for Eigenvalue Analysis

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

Parameter settings of eigenvalue analysis

##### 1.7.3.3.1.1 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)

##### 1.7.3.3.1.2 Example of Use

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

#### **1.7.3.4 Control Data for Heat Conduction Analysis**

##### **1.7.3.4.1 (1) !HEAT (4-1)**

Definition of control data regarding calculation

###### **1.7.3.4.1.1 Parameter**

N/A

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

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

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
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)

###### **1.7.3.4.1.2 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

```

##### **1.7.3.4.2 (2) !FIXTEMP (4-2)**

Definition of fixed temperature

###### **1.7.3.4.2.1 Parameter**

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

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

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

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

###### **1.7.3.4.2.2 Example of Use**

```

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

```

#### **1.7.3.4.3 (3) !CFLUX (4-3)**

Definition of centralized heat flux given to the node

##### **1.7.3.4.3.1 Parameter**

```
AMP = Flux history table name (specified in !AMPLITUDE)
** 2nd line or later **
(2nd line) NODE_GRP_NAME, Value
```

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

##### **1.7.3.4.3.2 Parameter**

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

#### **1.7.3.4.4 (4) !DFLUX (4-4)**

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

##### **1.7.3.4.4.1 Parameter**

```
AMP = Flux history table name (specified in !AMPLITUDE)
** 2nd line or later **
(2nd line) ELEMENT_GRP_NAME, LOAD_type, Value
```

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

##### **1.7.3.4.4.2 Parameter**

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

##### **1.7.3.4.4.2.1 Load Parameters**

<b>Load Type No.</b>	<b>Applied Surface</b>	<b>Parameter</b>
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

#### **1.7.3.4.5 (5) !SFLUX (4-5)**

Definition of distributed heat flux by surface group

#### **1.7.3.4.5.1 Parameter**

```
AMP = Flux history table name (specified in !AMPLITUDE)
** 2nd line or later **
(2nd line) SURFACE_GRP_NAME, Value
```

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
SURFACE_GRP_NAME	C	Surface group name
Value	R	Heat flux value

#### **1.7.3.4.5.2 Example of Use**

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

#### **1.7.3.4.6 (6) !FILM (4-6)**

Definition of heat transfer coefficient given to the boundary plane

#### **1.7.3.4.6.1 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
```

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
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

#### **1.7.3.4.6.2 Example of Use**

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

#### **1.7.3.4.6.2.1 Load Parameters**

<b>Load Type No.</b>	<b>Applied Surface</b>	<b>Parameter</b>
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

#### **1.7.3.4.7 (7) !SFILM (4-7)**

Definition of heat transfer coefficient by surface group

#### **1.7.3.4.7.1 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

#### 1.7.3.4.7.2 Example of Use

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

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

Definition of radiation factor given to boundary plane

##### 1.7.3.4.8.1 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

#### 1.7.3.4.8.2 Example of Use

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

#### 1.7.3.4.8.2.1 荷重パラメータ

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

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

Definition of radiation factor by surface group

##### 1.7.3.4.9.1 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

#### 1.7.3.4.9.2 Example of Use

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

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

Definition of weld line (Linear)

##### 1.7.3.4.10.1 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

#### 1.7.3.5 Control Data for Dynamic Analysis

##### 1.7.3.5.1 (1) DYNAMIC

Dynamic analysis control

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

##### 1.7.3.5.1.1 Parameter

TYPE = LINEAR : Linear dynamic analysis

```

NONLINEAR : Nonlinear dynamic analysis

** 2nd line or later **

(2nd line) idx_eqa, idx_resp

Parameter Name Attributions Contents
idx_eqa I Solution of equation of motion (Direct time integration)
          (Default: 1)
          1: Implicit method (Newmark- $\beta$  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)

```

1.7.3.5.1.1.1 idx\_resp=1 (Time history response analysis)

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

<b>Parameter Name Attributions</b>	<b>Contents</b>
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) ganma , beta

<b>Parameter Name Attributions</b>	<b>Contents</b>
ganma R	Parameter $\gamma$ of Newmark- $\beta$ method (Default: 0.5)
beta R	Parameter $\beta$ of Newmark- $\gamma$ method (Default: 0.25)

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

<b>Parameter Name Attributions</b>	<b>Contents</b>
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 $R_m$ of Rayleigh damping (Default: 0.0)
ray_k R	Parameter $R_k$ of Rayleigh damping (Default: 0.0)

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

<b>Parameter Name Attributions</b>	<b>Contents</b>
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)

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
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: Output 2: 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: Output 2: Output (Node base) 3: Output (Element base)

#### 1.7.3.5.1.2 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, 0
```

#### 1.7.3.5.1.2.1 idx\_resp=2 (Frequency response analysis)

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

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
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

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
t_start	R	Analysis start time
t_end	R	Analysis end time

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

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
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 R_m of Rayleigh damping (Default: 0.0)
ray_k	R	Parameter R_k of Rayleigh damping (Default: 0.0)

(6th line) nout, vistype, nodeout

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
nout	I	Results output interval in time domain Visuzalization type

vistype	I	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)		

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
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

#### 1.7.3.5.1.3 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
```

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

Definition of velocity boundary conditions

##### 1.7.3.5.2.1 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

<b>Parameter Name</b>	<b>Attributions</b>	<b>Contents</b>
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)

##### 1.7.3.5.2.2 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.

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

Definition of acceleration boundary conditions

##### 1.7.3.5.3.1 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)

##### 1.7.3.5.3.2 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.

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

Definition of coupled surface (Used only in coupled analysis)

##### 1.7.3.5.4.1 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 \*\*

(2行目) COUPLING\_SURFACE\_ID

Parameter Name Attributions	Contents
SURFACE_ID C	Surface group name

#### 1.7.3.5.4.2 Example of Use

```
!COUPLE , TYPE=1  
SCOUPLE1  
SCOUPLE2
```

#### 1.7.3.5.5 (5) !EIGENREAD (5-5)

Controlling the input file for frequency response analysis

##### 1.7.3.5.5.1 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

#### 1.7.3.5.5.2 Example of Use

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

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

Defining external forces applied in frequency response analysis

##### 1.7.3.5.6.1 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

#### 1.7.3.5.6.2 Example of Use

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

#### 1.7.3.6 Solver Control Data

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

Control of solver

Mandatory control data

#### 1.7.3.6.2 Parameter

```
METHOD = Method(CG, BiCGSTAB, GMRES, GPBiCG, DIRECT, DIRECTmkl, MUMPS)
DIRECT: Direct method other than contact analysis (serial processing only)
DIRECTmkl: Direct method by Intel MKL in contact analysis (serial processing only)
MUMPS : Parallel direct method by MUMPS
When any of direct methods is selected, the data lines will be disregarded.
In 1D and 2D problems, only CG, DIRECT and MUMPS are valid.
In shell problems, only DIRECT and MUMPS are valid.
Thread-parallel computation by OpenMP is available in iterative methods
for 3D problems.

PRECOND = Preconditioner(1, 2, 3, 5, 10, 11, 12)
1, 2      : (Block) SSOR
3         : (Block) Diagonal Scaling
5         : AMG by multigrid preconditioner package ML (experimental)
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 or Diagonal Scaling is recommended.

ITERLOG = Whether solver convergence history is output (YES/NO) (Default: NO)

TIMELOG = Whether solver computation time is output (YES/NO) (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
2: MPC-CG method
3: Explicit master-slave elimination (Default)
Valid only in 3D problems with iterative solvers.
(Penalty method is always used for direct solvers and 1D and 2D iterative
solvers, and MPC-CG method is always used for 4D and 6D iterative solvers.)

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.

** 2nd line or later **

(2nd line) NIER, iterPREmax, NREST, NCOLOR_IN
```

Parameter Name	Attributions	Contents
NIER	I	No. of iterations (Default: 100) No. of iteration of preconditioning based on Additive Schwarz (Default: 1)
iterPREmax	I	(recommended value : 1 for serial computation, parallel computation with Diagonal scaling, and serial/parallel computation of problems with

		MPC, 2 for other 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)
		(3rd line) RESID, SIGMA_DIAG, SIGMA

<b>Parameter Name Attributions</b>		<b>Contents</b>
RESID	R	Truncation error (Default: 1.0e-8)
SIGMA_DIAG	R	Scale factor for diagonal elements when computing preconditioning matrix (Default: 1.0)
SIGMA	R	Not used (Default: 0.0)

#### 1.7.3.6.2.1 Example of Use

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

### 1.7.3.7 Post Process (Visualization) Control Data

#### 1.7.3.7.1 (1) !VISUAL (P1-0)

Specifies the visualization method.

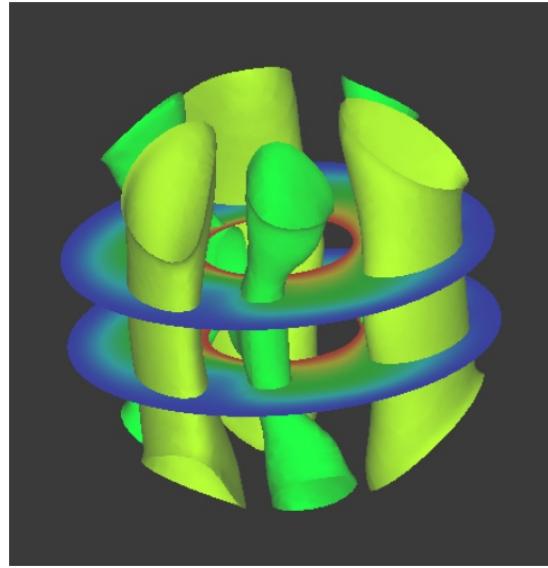
```
METHOD = PSR          : Surface rendering
  visual_start_step   : Specification of time step number which starts the visualization
process
                                (Default: 1)
  visual_end_step     : Specification of time step number which ends the visualization process
                                (Default: All)
  visual_interval_step : Specification of time step interval which performs the visualization
process
                                (Default: 1)
```

#### 1.7.3.7.2 (2) !surface\_num, !surface, !surface\_style (P1-1 - 3)

##### 1.7.3.7.2.1 !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.

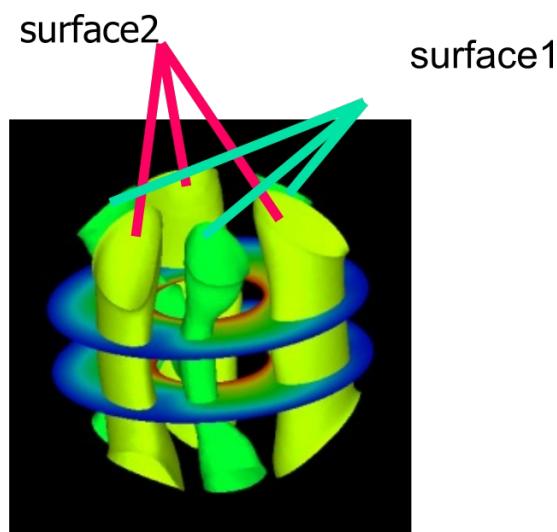


**Figure 7.4.1: Example of surface\_num Setting**

**1.7.3.7.2.2 !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
!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
```

#### 1.7.3.7.2.3 !surface\_style (P1-3)

Specifies the style of the surface.

1. Boundary plane
2. Isosurface
3. Arbitrary quadric surface  
coeff[1]x<sup>2</sup> + coeff[2]y<sup>2</sup> + coeff[3]z<sup>2</sup> + coeff[4]xy + coeff[5]xz  
+ coeff[6]yz + coeff[7]x + coeff[8]y + coeff[9]z + coeff[10]=0

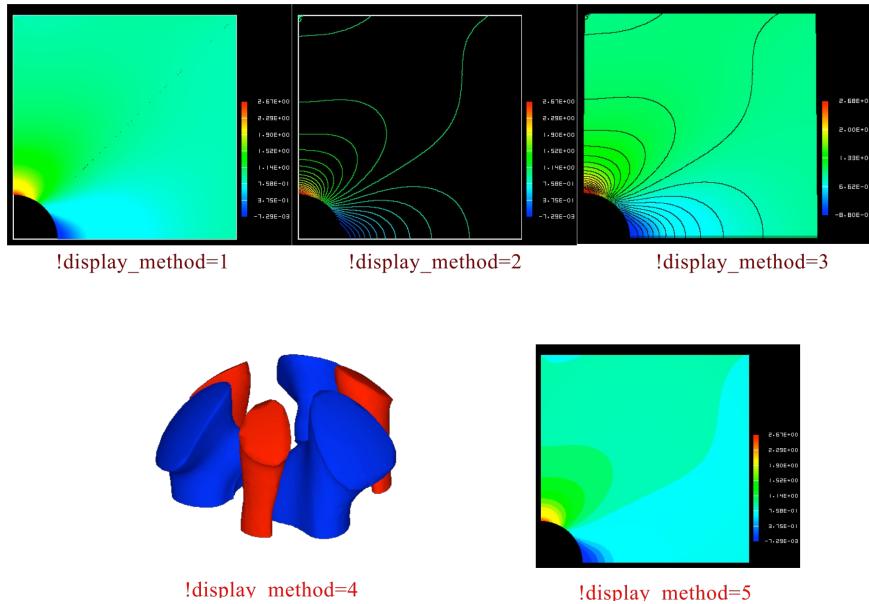


**Figure 7.4.3: Example of surface\_style Setting**

#### 1.7.3.7.3 (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**

#### 1.7.3.8 (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

##### 1.7.3.8.0.1 !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
```

#### 1.7.3.8.0.2 !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.

#### 1.7.3.8.0.3 !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

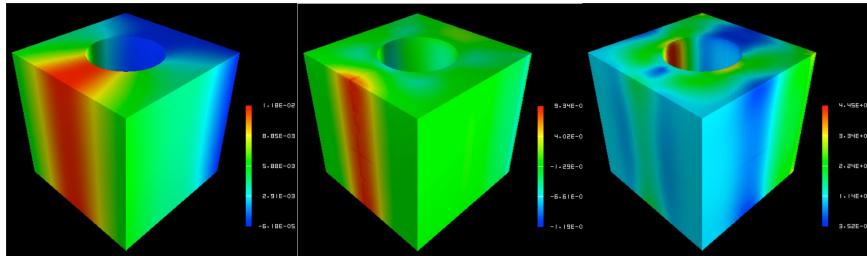
When !color\_comp\_name=STRAIN is specified  
1:  $\epsilon_x$ , 2:  $\epsilon_y$ , 3:  $\epsilon_z$   
4:  $\epsilon_{xy}$ , 5:  $\epsilon_{yz}$ , 6:  $\epsilon_{zx}$

When !color\_comp\_name=STRESS is specified  
1:  $\sigma_x$ , 2:  $\sigma_y$ , 3:  $\sigma_z$   
4:  $\tau_{xy}$ , 5:  $\tau_{yz}$ , 6:  $\tau_{zx}$

When !color\_comp\_name=TEMPERATURE is specified  
1: Temperature

In the structural analysis, for example;

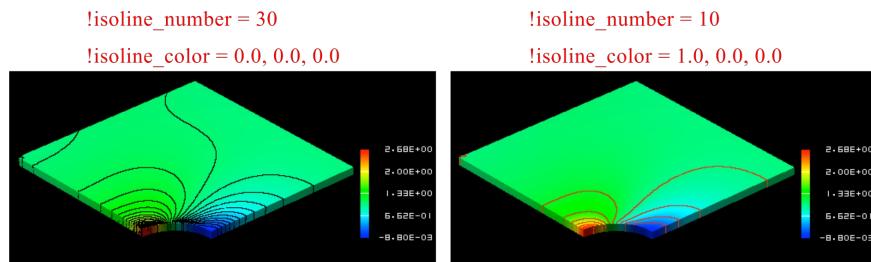
Physical Value	Displacement	Strain	Stress
No. of degrees of freedom	3	6	7



**Figure 7.4.5: Example of color\_comp, color\_subcomp and color\_comp\_name Setting**

#### 1.7.3.8.1 (5) !isoline\_number, !isoline\_color (P1-9, P2-22)

When display\_method=2,3 or 5



**Figure 7.4.6: Example of isoline\_number and isoline\_color Setting**

**1.7.3.8.2 (6) !initial\_style, !deform\_style (P1-15, P1-16)**

Specifies the display style of the initial shape and the deformed shape.

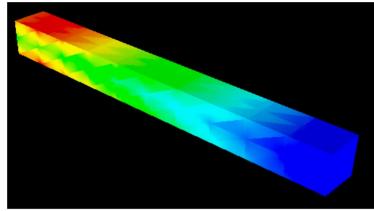
0. Not specified
1. Solid line mesh (Displayed in blue if not specified)
2. Gray filled pattern
3. Shading (Let the physical attributions respond to the color)
4. Dotted line mesh (Displayed in blue if not specified)

**1.7.3.8.2.1 (7) !deform\_scale (P1-14)**

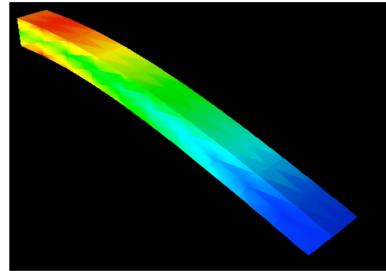
Specifies the displacement scale when displaying deformation.

Default: Auto

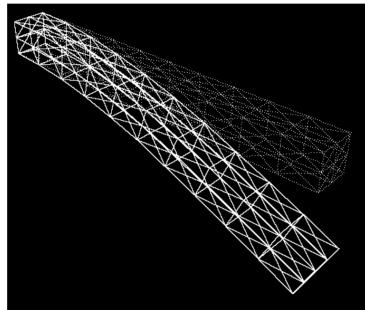
$$\text{standard\_scale} = 0.1 * \sqrt{x\_range^2 + y\_range^2 + z\_range^2} / \text{max\_deform}$$



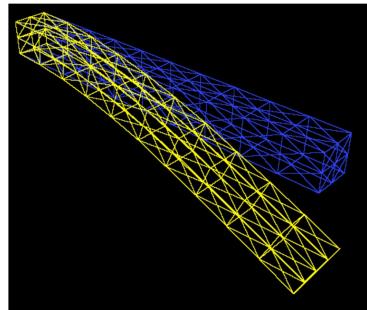
`!initial_style=2`  
`!deform_style = 0`



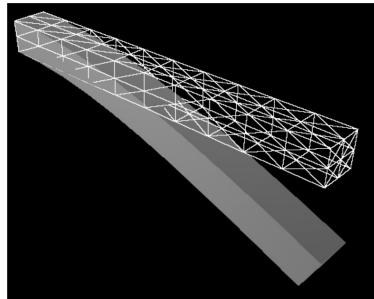
`!initial_style=0`  
`!deform_style = 2`



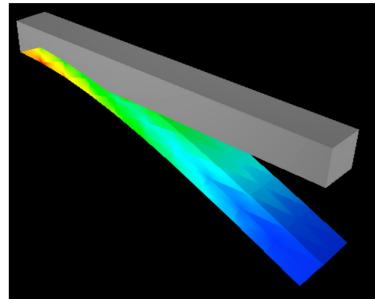
`!initial_style=4`  
`!deform_style = 1`  
`!initial_line_color = 1.0, 1.0, 1.0`



`!initial_style=1`  
`!deform_style = 1 NASTRAN style`  
`!initial_line_color = default`

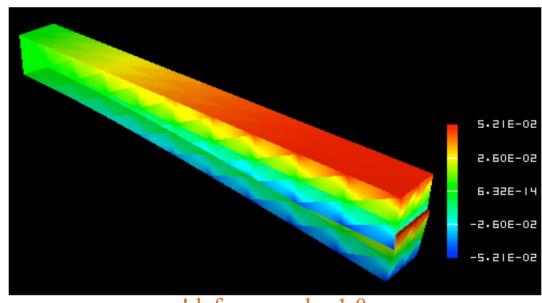


`!initial_style=1`  
`!deform_style = 2`

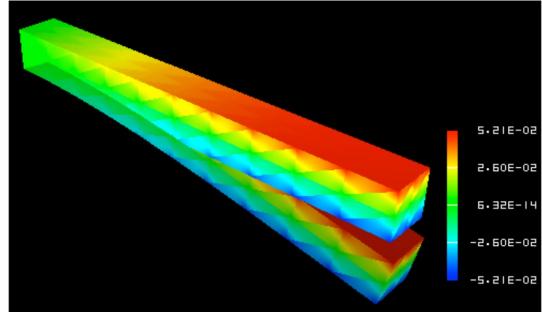


`!initial_style=2`  
`!deform_style = 3`

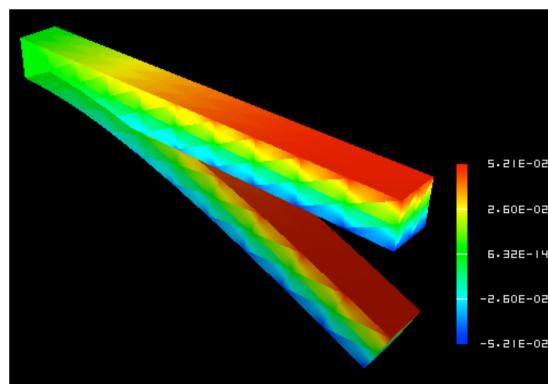
**Figure 7.4.7: Example of display\_styles Setting**



`!deform_scale=1.0`



`!deform_scale=2.0`



`!deform_scale=4.0`

**Figure 7.4.8: Example of deform\_scale Setting**

#### 1.7.3.8.2.2 (8) !output\_type (P1-19)

Specifies the type of output file. (Default: AVS)

AVS	: UCD data for AVS (only on object surface)
BMP	: Image data (BMP format)
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

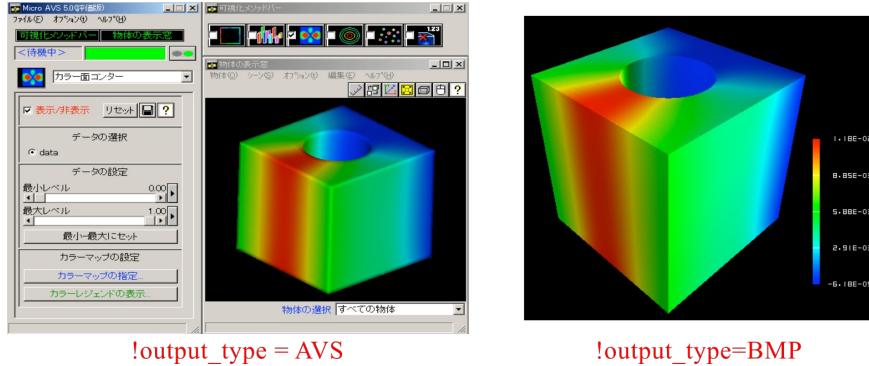


Figure 7.4.9: Example of output\_type

#### 1.7.3.8.3 (9) !x\_resolution, !y\_resolution (P2-1, P2-2)

Specifies the resolution when output\_type=BMP

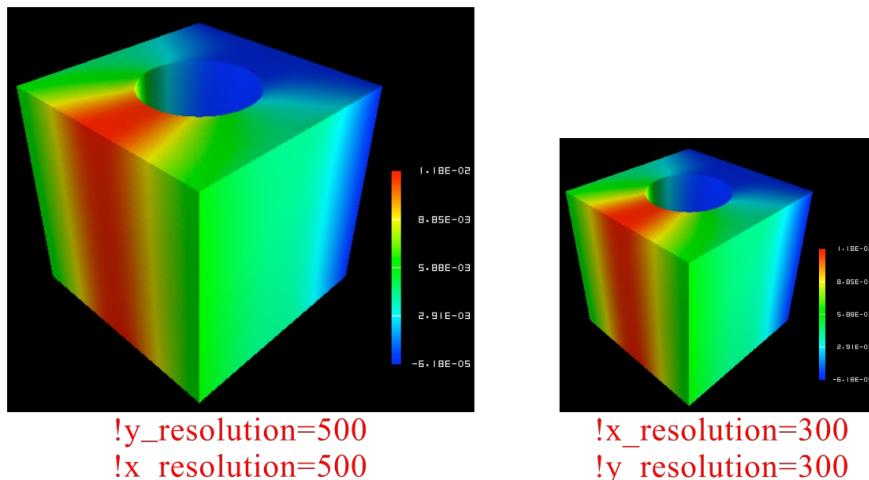


Figure 7.4.10: Example of x\_resolution and y\_resolution Setting

#### 1.7.3.8.4 (10) !viewpoint, !look\_at\_point, !up\_direction (P2-5, P2-6, P2-7)

##### 1.7.3.8.4.1 viewpoint

Specifies the viewpoint position by coordinates.

Default:  $x = (\text{xmin} + \text{xmax})/2.0$ ,  $y = \text{ymin} + 1.5 (\text{ymax} - \text{ymin})$ ,  $z = \text{zmin} + 1.5 (\text{zmax} - \text{zmin})$

##### 1.7.3.8.4.2 look\_at\_point

Specifies the look at point position.

Default: Center of data

##### 1.7.3.8.4.3 up\_direction

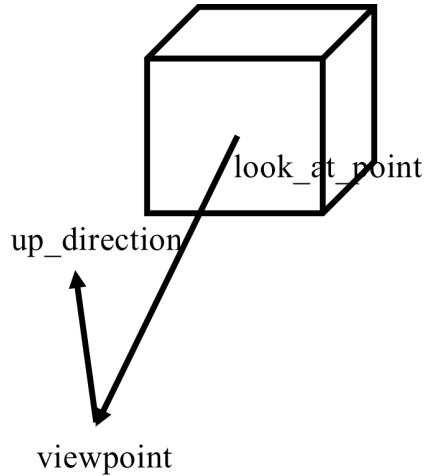
Specifies the view frame in viewpoint, look\_at\_point and up\_direction.

default: 0.0 0.0 1.0

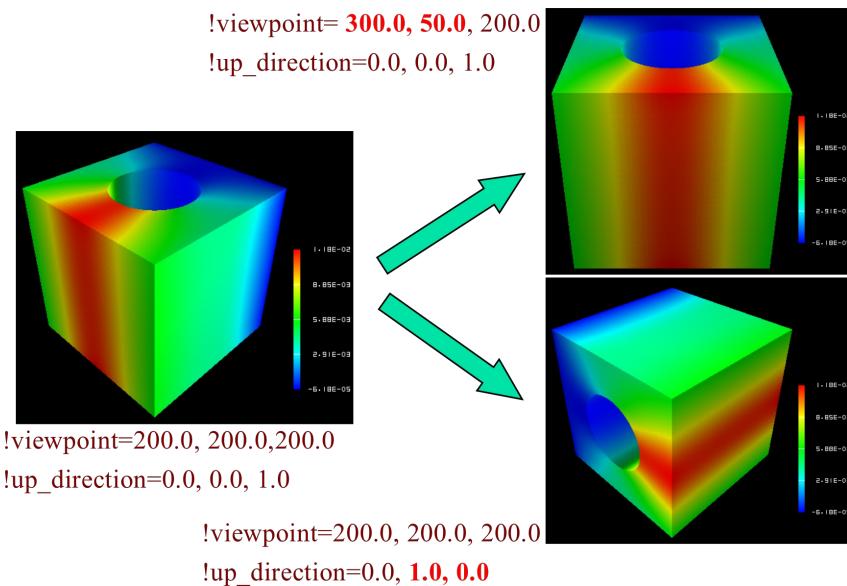
##### 1.7.3.8.4.4 View coordinate frame

- Origin: look\_at\_point

- Z-axis: viewpoint - look\_at\_point
- X-axis: up\_direction × z axis



**[Figure 7.4.11: View Frame Determination Method]**

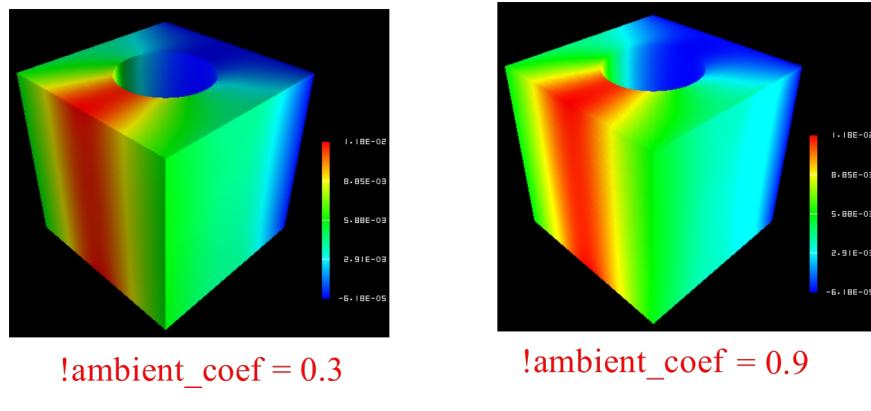


**Figure 7.4.12: Example of !viewpoint, look\_at\_point and up\_direction Setting**

#### 1.7.3.8.5 (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.



#### 1.7.3.8.6 (12) '!color\_mapping\_bar\_on' '!scale\_marking\_on' '!num\_of\_scales' (P2-16 P2-17 P2-18)

`!color_mapping_bar_on`

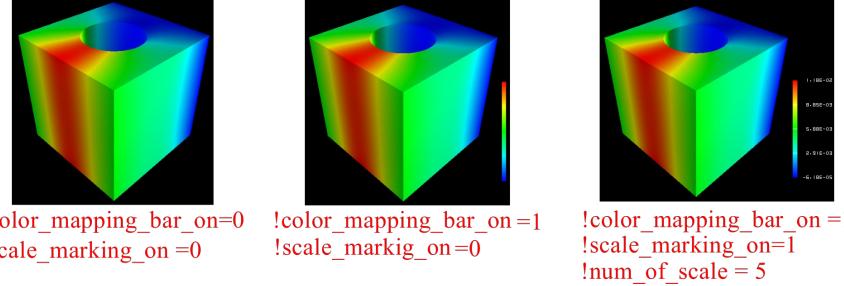
Specifies whether to display the color mapping bar.  
0: off 1: on (Default: 0)

`!scale_marking_on`

`color_mapping_bar`のメモリの有無を指定する  
0: off 1: on (省略値: 0)

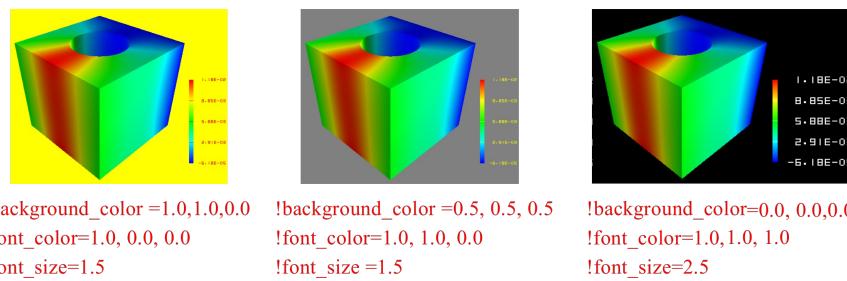
`!num_of_scales`

メモリの数を指定する。  
(省略値: 3)



#### 1.7.3.8.7 (13) !font\_size !font\_color !background\_color (P2-19 P2-20 P2-21)

Specifies the background color and character font.



**Figure 7.4.15: Example of Background and Font Setting**

#### 1.7.3.8.8 (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`.



`!data_comp_name=pressure`

`!data_comp_name=vorticity`

`!data_subcomp=3`

**Figure 7.4.16: Example of data\_comp, data\_subcomp and data\_comp\_name Setting**

#### 1.7.3.8.9 (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, 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.

## 1.8 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.

### 1.8.1 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

### 1.8.2 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`.

#### 1.8.2.1 (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

### 1.8.2.2 (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)

## 1.8.3 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.

### 1.8.3.1 (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

### 1.8.3.2 (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

## 1.8.4 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.

### 1.8.4.1 (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

```

#### **1.8.4.2 (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

```

### **1.8.5 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.

#### **1.8.5.1 (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.

#### **1.8.5.2 (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

#### **1.8.5.3 (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

## 1.9 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.

### 1.9.1 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

### 1.9.2 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.

#### 1.9.2.1 (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

#### 1.9.2.2 (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)

### 1.9.3 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.

#### 1.9.3.1 (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
```

#### 1.9.3.2 (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
```

### 1.9.4 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.

#### 1.9.4.1 (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
```

#### 1.9.4.2 (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

### 1.9.5 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.

#### 1.9.5.1 (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.

#### 1.9.5.2 (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

#### 1.9.5.3 (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

## 1.10 ステップ制御

英語に翻訳する

### 1.10.1 解析上の時間について

ここではFrontISTRの解析上の時間について、以下の通り用語の定義を行う：

- 現時刻：解析初期からの総経過時間
- ステップ時刻：ステップ開始からの経過時間
- 時間幅：ステップで解析する時間
- 相対時刻：時間幅に対するステップ開始からの経過時間の割合
- 時間増分：現時刻からつり合いを求める時刻までの増分

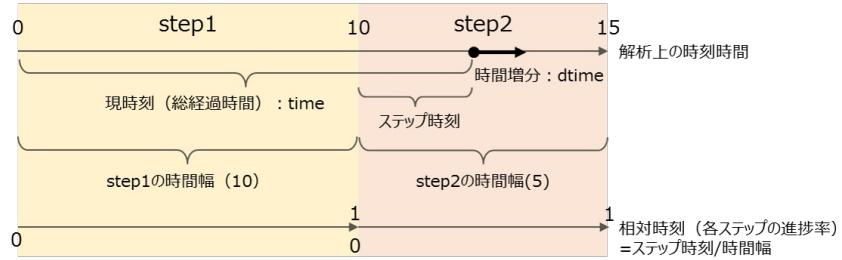


図4.7.1 解析上の時間に関する用語定義

### 1.10.2 静解析の制御

本開発コードにおいて、静解析は1つまたは複数の連続した解析ステップからなる。各解析ステップごとに基本境界条件の組を1つ与え、これを前述の増分解法によって解いていく。以下では解析ステップの反復をステップループ、増分解法の反復をサブステップループと呼ぶ。静解析の増分制御は、以下のいずれかから選択することができる。

- 固定増分による計算。収束に失敗した場合は直ちに計算を終了する。
- 自動増分・カットバックによる計算。収束状況に応じて増分量を変化させ、収束に失敗した場合には増分を小さくして再計算を行う。

#### 1.10.2.1 自動増分・カットバックの概要

自動増分・カットバックによる計算のフローは図4.7.2の通りである。

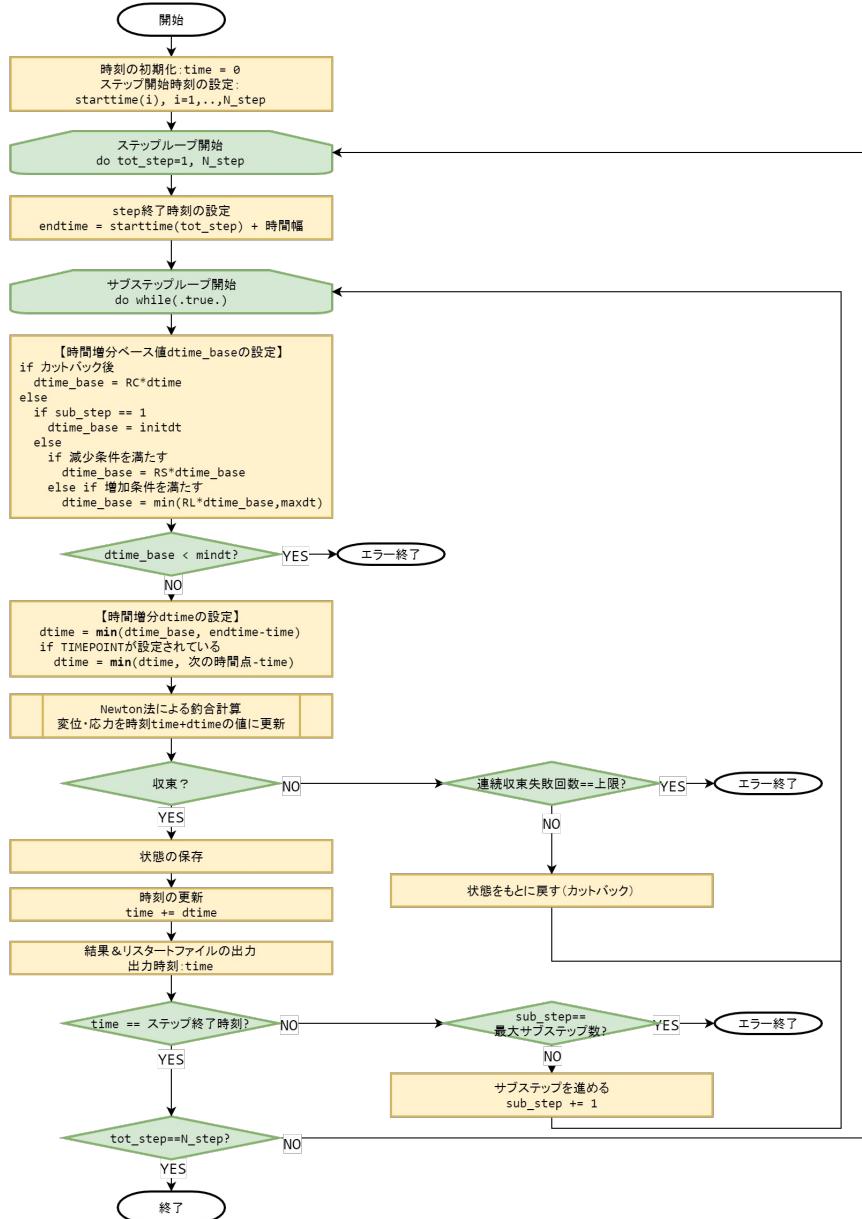


図4.7.2 増分割御フロー

フローの骨格は次の通りである。

1. ステップ1から最終ステップN\_stepまで、下記2.以降の手続きを繰り返す：
2. 時間増分の基準値dtime\_baseを、現在のdtime\_baseと前サブステップでの収束状況から定める。  
初回は初期時間増分initdtを用いる。
3. 実際の時間増分dtimeを、ステップ終了または直近の出力指定時刻までの残り時間とdtime\_baseの小さい方で定める。
4. 時刻time+dtimeでのつり合い計算を試みる
5. 収束に成功した場合は時刻をdtimeだけ進め、失敗した場合は時刻timeの状態を復元して2.に戻る。
6. timeがステップ終了時刻に到達したらステップを終了する

解析途中で下記に挙げるケースに該当した場合、非線形静解析の手続きは失敗と判断され、エラー終了する。

- timeがステップ終了時刻に到達する前にサブステップ数が上限に到達した場合
- 時間増分の基準値dtime\_baseが時間増分下限mindtを下回った場合
- 指定された N\_C 回連続して収束に失敗した場合

### 1.10.2.2 時間増分基準値dtime\_baseの調整

ステップ初回のdtime\_baseは、指定された初期時間増分initdtの値に設定される。それ以外の場合は、前のサブステップの収束状況に応じて次の通り設定される。

1. 収束に失敗した場合（カットバックされた場合）...dtimee\_baseにカットバック縮小率 R\_C を乗じた値
2. 収束に成功した場合
  1. 減少条件に該当する場合：dtime\_baseに減少率 R\_S を乗じた値
  2. 減少条件に該当せず、増加条件に該当する場合：dtime\_baseに増加率 R\_L を乗じた値と、時間増分上限maxdtの小さい方
  3. 減少条件にも増加条件にも該当しない場合：dtime\_baseは変化しない

### 1.10.2.3 増加・減少条件

自動増分調整機能では、増分を増加・減少させる条件を以下の変数を用いて判定する：

- N\_max : 前サブステップにおけるNewton法反復回数の最大値
- N\_sum : 前サブステップにおけるNewton法反復回数の合計値（接触反復が無い場合は N\_max に一致）
- N\_cont : 前サブステップにおける接触反復回数

減少条件は以下の両方が満たされるときである：

- N\_max, N\_sum, N\_cont の「いずれか一つ」が、各々の閾値NS\_max, NS\_sum, NS\_contを上回る
- 上記の状態が、N\_S 回以上連続したサブステップで満たされる

増加条件は以下の両方が満たされるときである：

- N\_max, N\_sum, N\_cont の「すべて」が、各々の閾値NL\_max, NL\_sum, NL\_cont以内である
- 上記の状態が、N\_L 回以上連続したサブステップで満たされる

### 1.10.2.4 計算および出力時刻の指定

自動時間増分は収束状況によって増分が変化するため、どの時刻につりあい計算および結果出力が行われるか事前に決定できない。これが不便である場合に、出力時刻のリストを与えることによって、希望する時刻におけるつり合い計算および結果出力を実行させることができる。出力時刻のリストが与えられたステップでは、指定された時刻にて必ず計算が行われるように、時間増分dtimeの値が調整される。

### 1.10.2.5 時間増分の使用方法

本機能に関する設定はすべて解析制御ファイルで行う。自動増分調整・カットバック機能は!STEPカードでTYPE\_INC=AUTOを指定することで有効になる。時間増分の調整関連のパラメータは、!AUTOINC\_PARAMを定義の上、!STEP, AUTOINCPARAMパラメータで各ステップごとに指定する。指定がない場合は!AUTOINC\_PARAMのデフォルトパラメータが使用される。出力時刻の指定は、!TIME\_POINTSで時刻リストの定義の上、!STEP, TIMEPOINTSパラメータで各ステップごとを行う。

#### 1.10.2.5.1 使用例

自動増分調整を有効にし、初期時間増分0.01、ステップ時間幅2.5、時間増分下限1E-5、時間増分上限0.3、最大サブステップ数を200に設定する。自動増分のパラメータセットはAP1（セット名）、出力時刻1.5, 2.7, 3.9を指定する。

```
!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
```

### 1.10.3 動解析の制御

本開発コードにおいて、動解析直接時間積分法は1つの解析ステップからなる。また動解析の増分制御は固定増分のみであり、自動時間増分機能を使用することはできない。