CAE-IF Recommended Practices Supplement

for

Non-Structural Mass in AP209

Version 1.3, September 3, 2025

Status: Final

<u>Contacts:</u>

CAE-IF	
Jochen Boy PROSTEP AG jochen.boy@prostep.com	

Technical	
Keith Hunten, P.E. (retired) Lockheed Martin Aeronautics Advanced Development Programs (retired) KAHunten@gmail.com	**Dmitry Krambalev** Jotne Connect Dmitry.Krambalev@jotne.com

Preface

This document is a supplement to the existing “Recommended Practices for AP 209 ed2 10303-209:2014” and “STEP AP209 ed2 Linear Static Structural FEA Handbook” documents. This supplement describes the use of non-structural mass in AP209 ed2, with examples from NASTRAN.

Please direct technical questions relating to this document to:

Keith A. Hunten

Tel: +1 (817) 637-5166

Email: KAHunten@gmail.com

Acknowledgements

The primary contributor to this supplement is Keith Hunten, working under contract with Jotne Connect, as part of their support for the PDES, Inc. membership organization. This work incorporates example data from Jotne’s EDMopenSimDM application and its AP209 translator software.

Document History

Revision	Section / Figure	Change
1.0		Initial document.
1.1		Updated cover page and page hears.
1.2	2.1.1	Updated the discussion on NSM and NSML1 cards to clarify that the VALUE of the mass in the NSM card is mass per unit area, and that the VALUE of the mass in the NSML1 card is total mass to be applied to one or many elements and/or element properties.
1.3		Including Jotne team review comments

Contents

Table of Contents

Figures

Tables

1 Introduction

This Recommended Practices document has been prepared as a usage guide for industry. This document assumes that the reader has at least a rudimentary knowledge of both 10303 STEP and its associated AP209 ed2 (10303-209 ed2) application domains.

2 Using AP209 ed2 to represent Non-Structural Mass

This section describes how AP209 ed2 is intended to be used to represent non-structural mass usage in finite element model. In particular this document focuses on how non-structural mass (NSM) is represented in element property representations, and how a one-dimensional point element representing a mass matrix is specified.

2.1 Non-Structural Mass in Element Property Representations

In AP209 non-structural mass is specified in the element property representations. AP209 has a single attribute for NSM for element properties. This was done as it represents the abstraction to the lowest common denominator of various FEA applications approaches to specifying non-structural mass. However, NASTRAN has the ability to apply one or many NSM input cards to an element property. This results in the requirement that if there are multiple NSM cards that reference an instance of an 1D or 2D element or element property card then they must be summed before applying the final sum to the property attribute of the AP209 element property instance.

A side effect of this difference in application of NSM is, there is an inability to exactly reproduce the original NASTRAN input deck in a circle test. However, the resulting analysis will be identical for the original input and regenerated input NASTRAN decks; the end result would be a single NSM card unlike the originally input.

2.1.1 Mapping the NSM, NSML1 *Cards to AP209*

In NASTRAN the NSM card (see Figure 1) has one or many pairs of an NSM value (VALUE) and either a reference to a Property or Element card (ID). These values will need to be summed to the target AP209 Curve and Surface Element Property instance ".non_structural_mass"-attribute (see Figure 3).

NOTE that the VALUE in the NSM card is a mass per unit area, as is the ".non_structural_mass"-attribute in AP209.

1	2	3	4	5	6	7	8	9	10
NSM	TYPE	ID	VALUE	ID	VALUE	ID	VALUE	ID	VALUE

Figure 1 NSM Card Format

```
*)
ENTITY surface_section
SUPERTYPE OF (ONEOF (uniform_surface_section,
uniform_surface_section_layered));
offset : measure_or_unspecified_value;
non_structural_mass : measure_or_unspecified_value;
non_structural_mass_offset : measure_or_unspecified_value;
END_ENTITY;
(*
```

Figure 2 Non-Structural Mass Attribute

```
*)
ENTITY curve_element_section_derived_definitions
SUBTYPE OF (curve_element_section_definition);
cross_sectional_area : context_dependent_measure;
```

```
shear_area : ARRAY [1:2] OF measure_or_unspecified_value;
second_moment_of_area : ARRAY [1:3] OF context_dependent_measure;
torsional_constant : context_dependent_measure;
warping_constant : measure_or_unspecified_value;
location_of_centroid : ARRAY [1:2] OF measure_or_unspecified_value;
location_of_shear_centre : ARRAY [1:2] OF measure_or_unspecified_value;
location_of_non_structural_mass
: ARRAY [1:2] OF measure_or_unspecified_value;
non_structural_mass : measure_or_unspecified_value;
polar_moment : measure_or_unspecified_value;
END_ENTITY;
(*
```

Figure 3 Curve Non-Structural Mass Attribute

The NASTRAN NSML1 card (see Figure 4) has one or many pairs of NSM value (VALUE) referencing an Element or Property card with several input range options.

NOTE that the VALUE in the NSML1 case is the total mass to be distributed amongst the various IDs of property and element cards, not the mass per unit area as used by the NSM card and the ".non_structural_mass”-attribute of an element property in AP209. Because of this the VALUE must be converted to mass per unit area before recording it in an AP209 representation.

A Property or Element ID may be specified by an Integer > 0, “ALL”, “THRU”, “BY”, or N (the BY increment) (see NASTRAN manual for more details). These range options govern the requirements for the summing of the NSM value to the target Element Property instance “.non_structural_mass”-attribute.

1	2	3	4	5	6	7	8	9	10
NSML1	TYPE	ID	VALUE	ID	ID	ID	ID	ID	etc.

Figure 4 NSML1 Card Format

2.1.2 Mapping the NSMADD Card to AP209

The NASTRAN NSMADD card (see Figure 5) is used to aggregate the NSM and NSML1 Cards to an element Set, which then specifies to which elements that the non-structural mass is applied. This set can contain NSM, NSML, NSM1, and NSML1 Cards. The Set is then selected by the Case Control NSM card.

1	2	3	4	5	6	7	8	9	10
NSMADD	SID	S1	S2	S3	S4	S5	S6	etc.	

Figure 5 NSMADD Card Format

NOTE that the NASTRAN Case Control NSM specifies which Load Case has the specified non-structural masses applied.

NOTE that if there are several NASTRAN CASE Control Load Cases, each with different NSM sets, then AP209 requires that the input NASTRAN deck shall be split into as many AP209 Models (for example, separate AP209 files) as there is no provision in AP209 to have more than one NSM for a given element property that is selectable by a Load Case (control_analysis_step).

2.1.3 Mapping the PBARL Card to AP209

The NASTRAN PBARL card (see Figure 6) is a 2D beam element specification that uses beam cross-section shape input data to calculate the various attributes of a beam cross-section required for stiffness calculation and stress recovery. This maps to the combination of the AP209 curve_element_section_derived_definition entity and the fea_curve_section_geometric_relationship entity that provides the cross-section shape information.

1	2	3	4	5	6	7	8	9	10
PBARL	PID	MID	GROUP	TYPE					
DIM1	DIM2	DIM3	DIM4	DIM5	DIM6	DIM7	DIM8		
DIM9	etc.	NSM							

Figure 6 PBARL Card Format

The three curve_element_section_derived_definition (see Figure 3) mandatory attributes (.cross_sectional_area, .second_moment_of_area, and . torsional_constant) are the minimum constants required to calculate a 6 DoF beam stiffness. This stiffness can be derived from the cross-section shape information (though not a simple calculation as it requires a complex algorithm to calculate the second moment and torsional constant). Not providing these attributes will cause a fatal error when trying to analyze the model. Human intervention to derive the named shape and the derived constants would be the simpler approach (probably the only approach as the PBARL card requires the cross-section name and associated dimensions).

2.2 Non-Structural Mass in Point Element Representations

In AP209 a point_element_representation (see Figure 7) instance is used to apply a mass matrix to a node (GRID in NASTRAN) in a finite element model. The NASTRAN equivalent is a CONM2 card. The mapping between these two representations can be quite complex due to the various options of coordinate frames for the node, the element, and the offset of the center of mass from the node/GRID. These options are illustrated in Figure 8.

```
*)
ENTITY point_element_representation
SUBTYPE OF (element_representation);
model_ref : fea_model;
description : text;
matrix_set UNIQUE
UR1: model_ref, SELF\representation.name;
WHERE
: SET [1:?] OF point_element_matrix;
WR1: SIZEOF (QUERY(item <* SELF\representation.items |
'STRUCTURAL_RESPONSE_REPRESENTATION_SCHEMA.' +
'FEA_AXIS2_PLACEMENT_3D'
IN TYPEOF (item))) = 1;
FU1: required_0d_nodes (
SELF\element_representation.node_list);
END_ENTITY;
(*

*)
ENTITY point_element_matrix
SUPERTYPE OF (ONEOF(stationary_mass,
grounded_spring,
grounded_damper));
END_ENTITY;
(*

*)
```

```
ENTITY stationary_mass
SUBTYPE OF (point_element_matrix);
mass : ARRAY [1:3] OF context_dependent_measure;
moments_of_inertia : symmetric_tensor2_3d;
coordinate_system : fea_axis2_placement_3d;
offset_vector : ARRAY [1:3] OF context_dependent_measure;
END_ENTITY;
(*
```

Figure 7 Point_element_representation and associated SUBTYPES

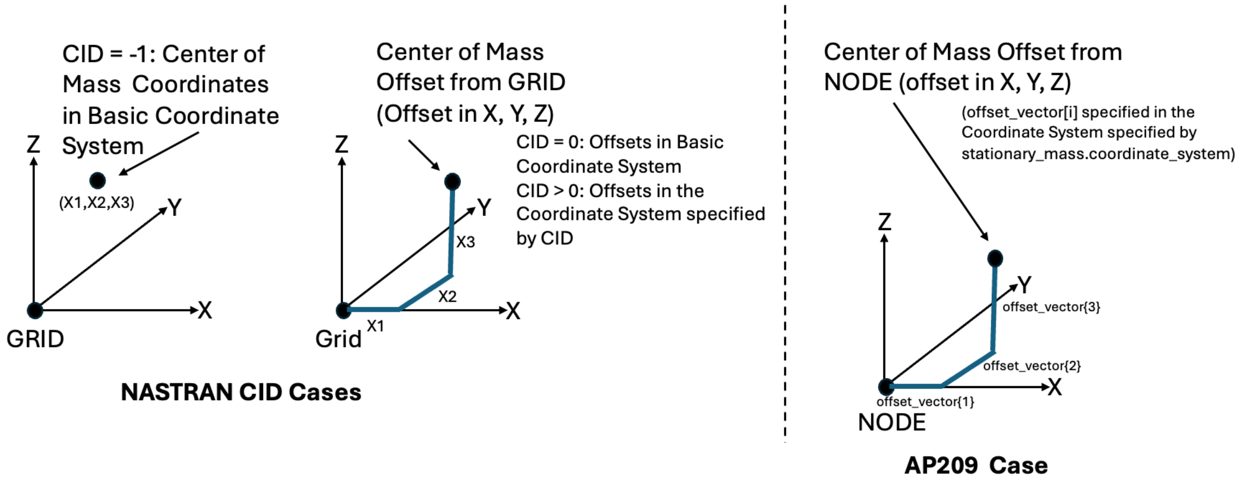

Figure 8 Offset Diagrams for NASTRAN and AP209

2.2.1 Coordinate Frames for Point Element Representations in AP209

The point_element_representation entity in AP209 has a WHERE-rule WR1 that specifies the coordinate frame for the Mass matrix:

WR1: there shall be exactly one coordinate system associated with a point_element_representation.

NOTE that a matrix coordinate system instance of FEA_AXIS2_PLACEMENT_3D is the coordinate system with respect to which all of the matrices of the point element are defined. The degrees of freedom at a node are defined with respect to the local orthogonal triad of the specified coordinate system at that node.

The stationary_mass subtype of the point_element_representation entity has two attributes that define the offset of the specified mass matrix:

> coordinate_system: the coordinate system which is used to specify the offset.
>
> offset_vector: the distance from the node to the position of the centre of mass with respect to the local orthogonal triad of the specified coordinate system at the node.

2.2.2 Coordinate Frames for CONM2 Elements in MSC NASTRAN

The NASTRAN CONM2 card (see Figure 9) has three CID options for specifying the offset of the non-structural mass matrix:

> CID = -1: in which case X1, X2, X3 are the coordinates, not Offsets, of the center of gravity of the mass in the basic coordinate system;
>
> The grid point locations may be defined in a non-basic coordinate system. In this case, the values of Iij must be in a coordinate system that parallels the basic coordinate system.
>
> CID = 0: in which case the X1, X2, X3 are the Offsets of the center of gravity of the mass in the basic coordinate system;

CID > 0: in which case the X1, X2, X3 are the Offsets of the center of gravity of the mass in the specified coordinate system CID.

1	2	3	4	5	6	7	8	9	10
CONM2	EID	G	CID	M	X1	X2	X3		
	I11	I21	I22	I31	I32	I33			

Figure 9 CONM2 Card Format

2.2.3 Mapping the CONM2 Card to AP209

The MSC NASTRAN CONM2 center of mass offset distances must be converted to an offset vector to be specified in AP209 (see Figure 8).

The AP209 offset_vector[i] is specified with respect to the coordinate frame specified in the stationary_mass attribute coordinate_system.

The mass matrix coordinate frame is specified by the Where Rule WR1 of the point_element_representation.

For CID = -1 or -0 the mass matrix is specified in the NASTRAN basic coordinate system, which maps to the asserted base coordinate frame in an AP209 file.

> In this case, the values of Iij must be in a coordinate system that parallels the basic coordinate system.
>
> To create the AP209 offset_vector[i] first cast the coordinates of the grid point G into the basic coordinate frame, then create the offset_vector[i] from the difference in the coordinate of the grid and the Offset X1, X2, X3. The coordinate_system in the point_element_representation stationary_mass instance must then be congruent to the asserted base coordinate system of the AP209 file.

For CID > 0 the mass matrix is specified in the coordinate frame CID.

For all CID cases the AP209 coordinate frame of the mass matrix is specified by the Where Rule WR1 in the point_element_representation instance.

> For CID = -1, 0 the fea_axis2_pacement_3d must be positioned at (0,0,0) and oriented congruent to the asserted base coordinate frame of the AP209 file;
>
> For CID > 0 the mass matrix coordinate frame is specified by an fea_axis2_pacement_3d specified by the Where Rule WR1 in the point_element_representation instance, with no restrictions in its orientation.

3 References and Links

NASTRAN: 2012.2 Quick Reference Guide, MSC Software

ISO 10303-209:2014: Multidisciplinary analysis and design, https://www.iso.org/standard/59780.html

AP209 ed2 Linear Static Structural FEA Handbook, Volume 1: ISO 10303-209 ed2 - Pilot Test Models 1-4, https://www.mbx-if.org/home/wp-content/uploads/2024/06/HandbookVolume1_V2.2_CAE-IF.pdf

LOTAR International	http://www.lotar-international.org
LOTAR EAS WG	http://www.lotar-international.org/lotar-workgroups/engineering-analysis-simulation.html
NAFEMS	http://www.nafems.org
AFNeT	http://www.afnet.fr
PDES, Inc.	https://www.pdesinc.org
prostep ivip	http://www.prostep.org/en.html
AIA	http://www.aia-aerospace.org
ASD-STAN	http://www.asd-stan.org
ISO STEP AP209 ed2	http://www.ap209.org
MBx-IF	https://www.mbx-if.org/home/mbx/
CAE-IF	https://www.mbx-if.org/home/cae/

Table 1 Useful Links for Further Reading on ISO 10303 (STEP)

Annex A Jotne Instantiation Examples

4 Example Model – Point Mass Element

The Conm2.bdf model is a simple synthetic test case to present a conversion of CONM2 Nastran card to AP209 entities.

Grid points and Local Coordinate system are reused from ATS pilot model, converted in the same way, and thus will not be described in detail here. Refer to that example in “STEP AP209 ed2 Linear Static Structural FEA Handbook - Volume 1: FEA Input for LOTAR EAS Pilot Study #1” for the details.

4.1 Element definition

CONM2 element defines a concentrated mass at Grid point in a form specialized for Rigid Body. This example doesn’t include any rigid bodies (Nastran cards RBE2, etc.) which in real-life models would be defined over the same reference Grid points where concentrated mass is defined. CONM2 element translates into instances of point_element_representation and stationary_mass entities.

4.1.1 FE Model Input

There are 3 Nastran elements defined using the CONM2 input card shown in Figure 1. They reference 3 different Grid points and use different local coordinate system modes to show effect on translation of offset vector.

1	2	3	4	5	6	7	8	9	10
CONM2	EID	G	CID	M	X1	X2	X3		
	I11	I21	I22	I31	I32	I33			

```
CONM2   1       1       0       0.1     3.1     3.2     3.3
*               2.1             2.2                     2.3
CONM2   2       2       1       0.2     3.1     3.2     3.3
*               2.1             2.2                     2.3
CONM2   3       3       -1      0.3     3.1     3.2     3.3
*               2.1             2.2                     2.3
```

Figure 10 Nastran input specification for CONM2

- *EID* is an ID of card
- *G* is the ID of referenced Grid point
- *CID* is the ID of coordinate system for both Iij and X1,X2,X3
- *M* is mass value
- *X1,X2,X3* is an offset vector from grid point to the center of gravity of the mass (or absolute coordinates of the center of gravity if CID=-1)
- *Iij* are mass moments of inertia, where components I21,I31,I32 have implied negative sign (i.e. these values stored in card are always positive, and negative sign is always applied before using them)

CID field of first CONM2 element is 0 (it could be blank as well) which means basic coordinate system is used to define offset distances X1, X2, X3 from grid to center of gravity of the mass.
CID field of second CONM2 element is 1 which means local coordinate system with ID 1 is used to define offset distances X1, X2, X3 (in this example it is the same coordinate system as used for all Grid points).
CID field of third CONM2 element is -1 which means X1, X2, X3 are actual coordinates of center of gravity of the mass in basic coordinate system.

4.1.2 EXPRESS schema

The Express schema entities and types used to map CONM2 are shown in Figure 2.

```
ENTITY point_element_representation
      SUBTYPE OF (element_representation);
         model_ref : fea_model;
         description : text;
         matrix_set : SET [1:?] OF point_element_matrix;
      UNIQUE UR1: model_ref, SELF\representation.name;
      WHERE WR1: SIZEOF (QUERY(itemF\element_representation.node_list);
END_ENTITY;

ENTITY stationary_mass
      SUBTYPE OF (point_element_matrix);
         mass : ARRAY [1:3] OF context_dependent_measure;
         moments_of_inertia : symmetric_tensor2_3d;
         coordinate_system : fea_axis2_placement_3d;
         offset_vector : ARRAY [1:3] OF context_dependent_measure;
END_ENTITY;

TYPE symmetric_tensor2_3d = SELECT
      (isotropic_symmetric_tensor2_3d,
      orthotropic_symmetric_tensor2_3d,
      anisotropic_symmetric_tensor2_3d);
END_TYPE;

TYPE anisotropic_symmetric_tensor2_3d =
      ARRAY [1:6] OF context_dependent_measure;
END_TYPE;
```

Figure 11 EXPRESS Schema fragment for CONM2 mapping

- *Matrix_set* is set to an instance of stationary_mass entity
- *Mass* is an array of mass values acting correspondingly in directions x, y, z
- *Moments_of_intertia* is moments of inertia of a point about center of mass. SELECT value type of the field is set to anisotropic_symmetric_tensor2_3d specialization to match Nastran input data
- *Coordinate_system* is a coordinate system used to specify offset
- *Offset_vector* is offset from the node to the position of the center of mass

4.1.3 STEP diagram

Main STEP instances and their relations used for mapping first CONM2 element card are shown in Figure 3.

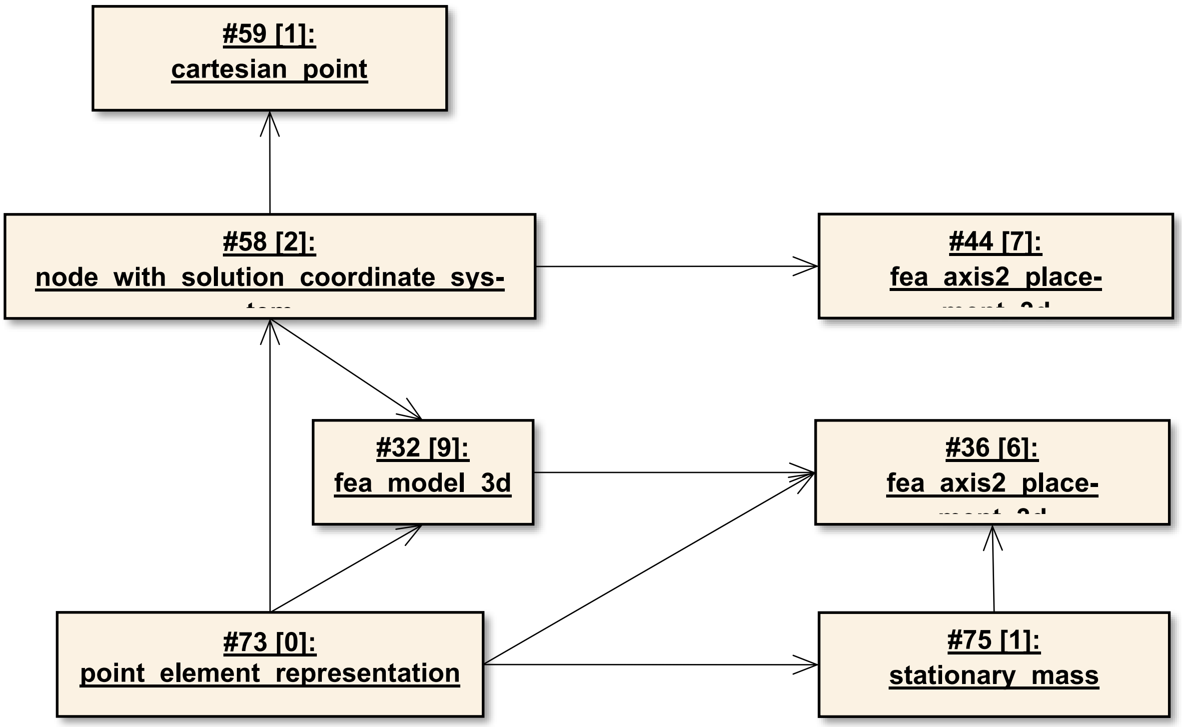

Figure 12 STEP instances diagram for CONM2

4.1.4 STEP Entities

CONM2 element maps to point_element_representation entity (provides element binding to target Grid/Node) which references stationary_mass entity (stores element matrix – with all parameters from CONM2). Moments of inertia are stored as anisotropic_symmetric_tensor2_3d type.

All Grid points and thus all Nodes are using their local coordinate system (in this example local coordinate system is named CSYS.1 in STEP, while basic coordinate system has CSYS.0 name in STEP).

Conversion of the first CONM2 element with CID=0 produces stationary_mass and point_element_representation both of which relate to CSYS.0 i.e. basic coordinate system as shown on Figure 4. Offset vector X1,X2,X3 from card is copied as is to offset_vector attribute of stationary_mass.

```
#44= FEA_AXIS2_PLACEMENT_3D('CSYS.1',#49,#47,#45,.CARTESIAN.,'');
#58= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('1',(#59,#44),#51,#32);

#36= FEA_AXIS2_PLACEMENT_3D('CSYS.0',#41,#39,#37,.CARTESIAN.,'');
#73= POINT_ELEMENT_REPRESENTATION('CONM2.1',(#36),#6,(#58),#32,(#75));


#75= STATIONARY_MASS((0.1,0.1,0.1),
ANISOTROPIC_SYMMETRIC_TENSOR2_3D((2.1,0.,0.,2.2,0.,2.3)),
```

```
#36,
(3.1,3.2,3.3));
```

Figure 13 STEP file fragment for CONM2, CID=0

Conversion of the second CONM2 element with CID=1 produces stationary_mass and point_element_representation both of which relate to CSYS.1 i.e. local coordinate system as shown on Figure 5. Offset vector X1,X2,X3 from card is copied as is to offset_vector attribute of stationary_mass.

```
#44= FEA_AXIS2_PLACEMENT_3D('CSYS.1',#49,#47,#45,.CARTESIAN.,'');
#63= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('2',(#64,#44),#51,#32);

#81= POINT_ELEMENT_REPRESENTATION('CONM2.2',(#44),#6,(#63),#32,(#83));

#83= STATIONARY_MASS((0.2,0.2,0.2),
ANISOTROPIC_SYMMETRIC_TENSOR2_3D((2.1,0.,0.,2.2,0.,2.3)),
#44,
(3.1,3.2,3.3));
```

Figure 14 STEP file fragment for CONM2, CID=1

Conversion of the third CONM2 element with CID=-1 produces stationary_mass and point_element_representation both of which relate to CSYS.0 i.e. basic coordinate system as shown on Figure 6. In this case coordinates of Grid point are converted into cartesian coordinates (if it they were not cartesian), then transformed into basic coordinate system and are subtracted from values X1,X2,X3 – the result is then stored in offset_vector attribute of stationary_mass.

```
#44= FEA_AXIS2_PLACEMENT_3D('CSYS.1',#49,#47,#45,.CARTESIAN.,'');
#67= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('3',(#68,#44),#51,#32);

#36= FEA_AXIS2_PLACEMENT_3D('CSYS.0',#41,#39,#37,.CARTESIAN.,'');
#89= POINT_ELEMENT_REPRESENTATION('CONM2.3',(#36),#6,(#67),#32,(#91));

#91= STATIONARY_MASS((0.3,0.3,0.3),
ANISOTROPIC_SYMMETRIC_TENSOR2_3D((2.1,0.,0.,2.2,0.,2.3)),
#36,
(1.1,5.2,2.3));
```

Figure 15 STEP file fragment for CONM2, CID=-1

4.1.5 Offset Diagrams for NASTRAN and AP209

Geometric interpretations of Nastran X1,X2,X3 vector depending on CID Nastran card field are shown in Figure 7.

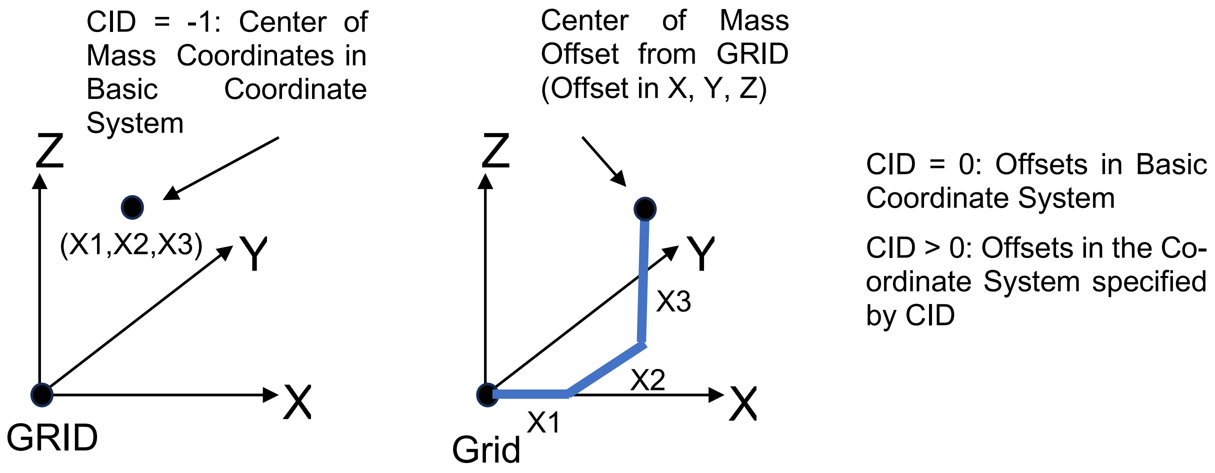

Figure 16 NASTRAN CID Cases

Geometric interpretation of AP209 offset vector is shown in Figure 8.

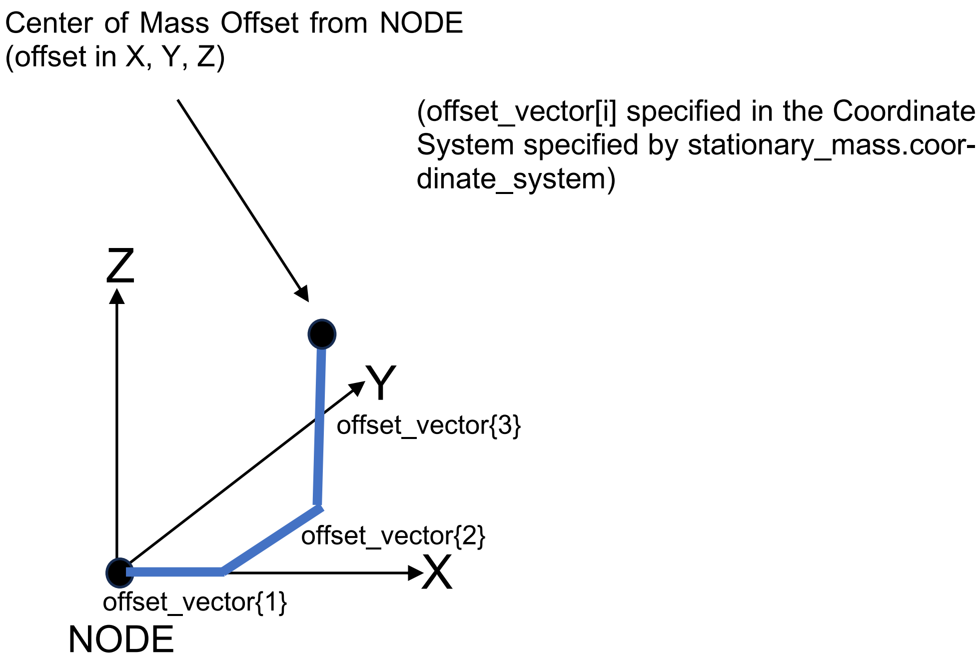

Figure 17 AP209 Case

4.2 CONM2 NASTRAN Deck

```
$ Linear Static Analysis
SOL 101
TIME 600
CEND
$
SEALL = ALL
```

```
SUPER = ALL
TITLE = Nastran job test case
ECHO = NONE
MAXLINES = 999999999
GPFORCE(PUNCH) = ALL
$
BEGIN BULK
PARAM   POST    -1
PARAM   AUTOSPC YES
PARAM   NOCOMPS -1
PARAM   PRTMAXIM YES
PARAM   GRDPNT  0
$
CONM2    1       1               0.1     3.1     3.2     3.3
*        2.1             2.2                     2.3
CONM2    2       2       1       0.2     3.1     3.2     3.3
*        2.1             2.2                     2.3
CONM2    3       3      -1       0.3     3.1     3.2     3.3
*        2.1             2.2                     2.3
$
GRID     1       1       0.     -2.      1.
GRID     2       1       1.     -2.      1.
GRID     3       1       2.     -2.      1.
$
CORD2R   1               0.      0.      0.      0.      0.      1.
         1.      0.      0.
$
ENDDATA
```

4.3 CONM2 AP209 File

```
ISO-10303-21;
HEADER;

/***************************************************************************
***************
* STEP Physical File produced by: EDMsix Version 602.34.151 Jun 25 2025
* Module:                         EDMstepFileFactory/EDMstandAlone
* Creation date:                  20250821 14:19:18
* Host:                           notebook
* Database:                       db
* Database version:               6007
* Database creation date:         Mon May  5 16:16:38 2025
* Schema:                         AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DE-
SIGN_MIM_LF
* Model:                          DataRepository.Fem_Model
* Model creation date:            Mon May  5 16:16:43 2025
* Header model:
* EDMuser:                        sdai-user
* EDMgroup:                       sdai-group
* License ID and type:            Test license : 10006  Expiry date: 2026-
01-10
* EDMstepFileFactory options:     0420000
****************************************************************************
**************/

FILE_DESCRIPTION((''),'2;1');
```

```
FILE_NAME('conm2.bdf.stp','2025-08-21T14:19:18',('ANONYMOUS USER'),('ANONY-
MOUS ORGANISATION'),'EDMsix Version 602.34.151 Jun 25 2025','','');
FILE_SCHEMA(('AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF'));
ENDSEC;

DATA;
#11= APPLICATION_CONTEXT('AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DE-
SIGN_MIM_LF');
#13= APPLICATION_PROTOCOL_DEFINITION('international standard','ap209_multi-
disciplinary_analysis_and_design',2011,#11);
#23= APPLIED_IDENTIFICATION_ASSIGNMENT('default',#24,(#17,#18,#19));
#25= APPLIED_ORGANIZATION_ASSIGNMENT(#26,#27,(#23));
#41= CARTESIAN_POINT('',(0.,0.,0.));
#49= CARTESIAN_POINT('',(0.,0.,0.));
#59= CARTESIAN_POINT('1',(0.,-2.,1.));
#64= CARTESIAN_POINT('2',(1.,-2.,1.));
#68= CARTESIAN_POINT('3',(2.,-2.,1.));
#97= CONTROL(#32,'control','FEMconverter','NASTRAN JOB TEST
CASE',(''),('NASTRAN'));
#37= DIRECTION('refX',(1.,0.,0.));
#39= DIRECTION('AxisZ',(0.,0.,1.));
#45= DIRECTION('refX',(1.,0.,0.));
#47= DIRECTION('AxisZ',(0.,0.,1.));
#36= FEA_AXIS2_PLACEMENT_3D('CSYS.0',#41,#39,#37,.CARTESIAN.,'');
#44= FEA_AXIS2_PLACEMENT_3D('CSYS.1',#49,#47,#45,.CARTESIAN.,'');
#32= FEA_MODEL_3D('FEM analysis',(#36),#6,'',(''),'');
#31= FEA_MODEL_DEFINITION('','',#30,.F.);
#6= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */(GEO-
METRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CON-
TEXT((#1,#2,#3,#4,#5))REPRESENTATION_CONTEXT('GLOBAL FEA CONTEXT','3d'));
#51= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */(GE-
OMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CON-
TEXT((#1,#2,#3,#4,#5))REPRESENTATION_CONTEXT('CSYS.1','3d'));
#9= /* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT
*/(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CON-
TEXT()REPRESENTATION_CONTEXT('default parametric context','3D'));
#24= IDENTIFICATION_ROLE('default-role',$);
#71= ITEM_DEFINED_TRANSFORMATION('',$,#44,#36);
#1= /* LENGTH_UNIT+SI_UNIT */(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT(.MILLI.,.ME-
TRE.));
#4= /* MASS_UNIT+SI_UNIT */(MASS_UNIT()NAMED_UNIT(*)SI_UNIT(.KILO.,.GRAM.));
#55= NODE_SET('node_set 1',(#58,#63,#67));
#58= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('1',(#59,#44),#51,#32);
#63= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('2',(#64,#44),#51,#32);
#67= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('3',(#68,#44),#51,#32);
#26= ORGANIZATION('JOTNE','JOTNE','');
#27= ORGANIZATION_ROLE('');
#2= /* PLANE_ANGLE_UNIT+SI_UNIT */(NAMED_UNIT(*)PLANE_AN-
GLE_UNIT()SI_UNIT($,.RADIAN.));
#73= POINT_ELEMENT_REPRESENTATION('CONM2.1',(#36),#6,(#58),#32,(#75));
#81= POINT_ELEMENT_REPRESENTATION('CONM2.2',(#44),#6,(#63),#32,(#83));
#89= POINT_ELEMENT_REPRESENTATION('CONM2.3',(#36),#6,(#67),#32,(#91));
#54= POINT_REPRESENTATION('point_rep 1',(#44,#55),#51);
#17= PRODUCT('/NULL','NASTRAN analysis','',(#14));
#14= PRODUCT_CONTEXT('/NULL',#11,'');
#19= PRODUCT_DEFINITION('','',#18,#16);
#16= PRODUCT_DEFINITION_CONTEXT('/NULL',#11,'analysis');
#18= PRODUCT_DEFINITION_FORMATION('default-id','',#17);
#30= PRODUCT_DEFINITION_SHAPE('mesh shape',$,#19);
```

```
#21= PRODUCT_RELATED_PRODUCT_CATEGORY('linear_static_analysis','',(#17));
#72= REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION('','',#54,#32,#71);
#56= SHAPE_DEFINITION_REPRESENTATION(#30,#54);
#3= /* SI_UNIT+SOLID_ANGLE_UNIT */(NAMED_UNIT(*)SI_UNIT($,.STERA-
DIAN.)SOLID_ANGLE_UNIT());
#5= /* SI_UNIT+THERMODYNAMIC_TEMPERATURE_UNIT */(NAMED_UNIT(*)SI_UNIT($,.DE-
GREE_CELSIUS.)THERMODYNAMIC_TEMPERATURE_UNIT());
#100= SPECIFIED_STATE('default input state','default input state');
#75= STATIONARY_MASS((0.1,0.1,0.1),ANISOTROPIC_SYMMETRIC_TEN-
SOR2_3D((2.1,0.,0.,2.2,0.,2.3)),#36,(3.1,3.2,3.3));
#83= STATIONARY_MASS((0.2,0.2,0.2),ANISOTROPIC_SYMMETRIC_TEN-
SOR2_3D((2.1,0.,0.,2.2,0.,2.3)),#44,(3.1,3.2,3.3));
#91= STATIONARY_MASS((0.3,0.3,0.3),ANISOTROPIC_SYMMETRIC_TEN-
SOR2_3D((2.1,0.,0.,2.2,0.,2.3)),#36,(1.1,5.2,2.3));
#33= STRUCTURAL_RESPONSE_PROPERTY('','',#31);
#34= STRUCTURAL_RESPONSE_PROPERTY_DEFINITION_REPRESENTATION(#33,#32);
ENDSEC;

END-ISO-10303-21;
```

5 Example Model – PBARL Case

5.1 Beam Element Property Definition

The PBARL example model is a simple synthetic test case to present a conversion of PBARL NASTRAN card to AP209 entities. This example of PBARL card is a conversion (with NSM > 0 to trigger use of CURVE_ELEMENT_SECTION_DERIVED_DEFINITIONS) and calculation of required missing parameters).

The BDF file in 4.2 is made from ATS2m5.bdf by only replacing the PBAR card with PBARL card, so refer to that example in “STEP AP209 ed2 Linear Static Structural FEA Handbook - Volume 1: FEA Input for LOTAR EAS Pilot Study #1”.

5.2 PBARL NASTRAN Deck

```
$ Linear Static Analysis
SOL 101
TIME 600
CEND
$
SEALL = ALL
SUPER = ALL
TITLE = Nastran EAS test case ATS2m5
ECHO = NONE
MAXLINES = 999999999
GPFORCE(PUNCH) = ALL
SUBCASE 1
   SUBTITLE=subcase1
   SPC = 100
   LOAD = 200
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
   STRAIN(PUNCH)=ALL
```

```
   FORCE(PUNCH)=ALL
SUBCASE 2
   SUBTITLE=subcase2
   SPC = 10
   LOAD = 22
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
   STRAIN(PUNCH)=ALL
   FORCE(PUNCH)=ALL
SUBCASE 3
   SUBTITLE=subcase3
   SPC = 10
   LOAD = 23
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
   STRAIN(PUNCH)=ALL
   FORCE(PUNCH)=ALL
$
BEGIN BULK
PARAM    POST     -1
PARAM    AUTOSPC  YES
PARAM    NOCOMPS  -1
PARAM    PRTMAXIM YES
PARAM    GRDPNT   0
$
PBARL    1        1                BAR
         2.       4.       0.1
$
CBAR*    1                1                1                2
*        0.               7.54979-8        1.
CBAR*    2                1                2                3
*        0.               7.54979-8        1.
CBAR*    3                1                3                4
*        0.               7.54979-8        1.
CBAR*    4                1                4                5
*        0.               7.54979-8        1.
CBAR*    5                1                5                6
*        0.               7.54979-8        1.
CBAR*    6                1                6                7
*        0.               7.54979-8        1.
CBAR*    7                1                7                8
*        0.               7.54979-8        1.
CBAR*    8                1                8                9
*        0.               7.54979-8        1.
CBAR*    9                1                9                10
*        0.               7.54979-8        1.
CBAR*    10               1                10               11
*        0.               7.54979-8        1.
CBAR*    11               1                11               12
*        0.               7.54979-8        1.
CBAR*    12               1                12               13
*        0.               7.54979-8        1.
CBAR*    13               1                13               14
*        0.               7.54979-8        1.
CBAR*    14               1                14               15
*        0.               7.54979-8        1.
CBAR*    15               1                15               16
```

```
*        0.              7.54979-8       1.
CBAR*    16              1               16              17
*        0.              7.54979-8       1.
$
MAT1     1       1.+7            .33     2.54-4  1.3-5   70.
$
GRID     1       1       0.      -2.     1.
GRID     2       1       1.      -2.     1.
GRID     3       1       2.      -2.     1.
GRID     4       1       3.      -2.     1.
GRID     5       1       4.      -2.     1.
GRID     6       1       5.      -2.     1.
GRID     7       1       6.      -2.     1.
GRID     8       1       7.      -2.     1.
GRID     9       1       8.      -2.     1.
GRID     10      1       9.      -2.     1.
GRID     11      1       10.     -2.     1.
GRID     12      1       11.     -2.     1.
GRID     13      1       12.     -2.     1.
GRID     14      1       13.     -2.     1.
GRID     15      1       14.     -2.     1.
GRID     16      1       15.     -2.     1.
GRID     17      1       16.     -2.     1.
$
SPCADD   10      100
$
LOAD     22      1.      1.      300     1.      400
LOAD     23      1.      1.      200     1.      300     1.      400
$
SPC1     100     123456  1
$
FORCE    200     17      0       1000.  -1.      0.      0.
FORCE    300     11      0       10.     0.     -1.      0.
FORCE    300     17      0       10.     0.     -1.      0.
FORCE    400     12      0       20.     0.     -1.      0.
FORCE    400     13      0       20.     0.     -1.      0.
FORCE    400     14      0       20.     0.     -1.      0.
FORCE    400     15      0       20.     0.     -1.      0.
FORCE    400     16      0       20.     0.     -1.      0.
$
CORD2R   1               0.      0.      0.      0.      0.      1.
         1.      0.      0.
$
ENDDATA
```

5.3 PBARL AP209 File

```
ISO-10303-21;
HEADER;

/****************************************************************************
***************
* STEP Physical File produced by: EDMsix Version 602.34.151 Jun 25 2025
* Module:                         EDMstepFileFactory/EDMstandAlone
* Creation date:                  20250822 14:18:35
* Host:                           dmitrynotebook4
* Database:                       W:\simdm\output\x64\Debug/db\d
* Database version:               6007
```

```
* Database creation date:         Mon May  5 16:16:38 2025
* Schema:                         AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DE-
SIGN_MIM_LF
* Model:                          DataRepository.Fem_Model
* Model creation date:            Mon May  5 16:16:43 2025
* Header model:
* EDMuser:                        sdai-user
* EDMgroup:                       sdai-group
* License ID and type:            Test license : 10006  Expiry date: 2026-
01-10
* EDMstepFileFactory options:     0420000
****************************************************************************
**************/

FILE_DESCRIPTION((''),'2;1');
FILE_NAME('pbarl.bdf.stp','2025-08-22T14:18:35',('ANONYMOUS USER'),('ANONY-
MOUS ORGANISATION'),'EDMsix Version 602.34.151 Jun 25 2025','','');
FILE_SCHEMA(('AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF'));
ENDSEC;

DATA;
#170= ANALYSIS_ITEM_WITHIN_REPRESENTATION('rectangular section','section to
geometry relation',#171,#172);
#11= APPLICATION_CONTEXT('AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DE-
SIGN_MIM_LF');
#13= APPLICATION_PROTOCOL_DEFINITION('international standard','ap209_multi-
disciplinary_analysis_and_design',2011,#11);
#23= APPLIED_IDENTIFICATION_ASSIGNMENT('default',#24,(#17,#18,#19));
#25= APPLIED_ORGANIZATION_ASSIGNMENT(#26,#27,(#23));
#173= AXIS2_PLACEMENT_2D('section-center',#176,#174);
#41= CARTESIAN_POINT('',(0.,0.,0.));
#49= CARTESIAN_POINT('',(0.,0.,0.));
#59= CARTESIAN_POINT('1',(0.,-2.,1.));
#64= CARTESIAN_POINT('2',(1.,-2.,1.));
#68= CARTESIAN_POINT('3',(2.,-2.,1.));
#72= CARTESIAN_POINT('4',(3.,-2.,1.));
#76= CARTESIAN_POINT('5',(4.,-2.,1.));
#80= CARTESIAN_POINT('6',(5.,-2.,1.));
#84= CARTESIAN_POINT('7',(6.,-2.,1.));
#88= CARTESIAN_POINT('8',(7.,-2.,1.));
#92= CARTESIAN_POINT('9',(8.,-2.,1.));
#96= CARTESIAN_POINT('10',(9.,-2.,1.));
#100= CARTESIAN_POINT('11',(10.,-2.,1.));
#104= CARTESIAN_POINT('12',(11.,-2.,1.));
#108= CARTESIAN_POINT('13',(12.,-2.,1.));
#112= CARTESIAN_POINT('14',(13.,-2.,1.));
#116= CARTESIAN_POINT('15',(14.,-2.,1.));
#120= CARTESIAN_POINT('16',(15.,-2.,1.));
#124= CARTESIAN_POINT('17',(16.,-2.,1.));
#176= CARTESIAN_POINT('',(-1.,-2.));
#130= CHARACTERIZED_OBJECT('property',$);
#566= CONTROL(#32,'control','FEMconverter','NASTRAN EAS TEST CASE
ATS2M5',(''),('NASTRAN'));
#570= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#566,'SUBCASE1',1,#569,'load
case',#571);
#621= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#566,'SUBCASE2',2,#569,'load
case',#622);
#670= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#566,'SUBCASE3',3,#569,'load
case',#671);
```

```
#571= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('1','SUBCASE1',#572);
#622= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('2','SUBCASE2',#623);
#671= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('3','SUBCASE3',#672);
#607= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#594,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_SCALAR_VARIABLE(.CURVE_AXIAL_FORCE.),$);
#618= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#594,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_SCALAR_VARIABLE(.TORQUE.),$);
#619= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#594,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_VECTOR_2D_VARIABLE(.CURVE_SHEAR_FORCE.),$);
#662= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#655,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_SCALAR_VARIABLE(.CURVE_AXIAL_FORCE.),$);
#667= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#655,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_SCALAR_VARIABLE(.TORQUE.),$);
#668= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#655,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_VECTOR_2D_VARIABLE(.CURVE_SHEAR_FORCE.),$);
#694= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#687,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_SCALAR_VARIABLE(.CURVE_AXIAL_FORCE.),$);
#699= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#687,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_SCALAR_VARIABLE(.TORQUE.),$);
#700= CURVE_3D_ELEMENT_CONSTANT_SPECIFIED_VARIABLE_VALUE(#687,#215,UNSPECI-
FIED_VALUE(.UNSPECIFIED.),CURVE_VECTOR_2D_VARIABLE(.CURVE_SHEAR_FORCE.),$);
#215= CURVE_3D_ELEMENT_DESCRIPTOR(.LINEAR.,'LINEAR_CURVE.BAR',((ENUMER-
ATED_CURVE_ELEMENT_PURPOSE(.AXIAL.)),(ENUMERATED_CURVE_ELEMENT_PUR-
POSE(.Y_Y_BENDING.),ENUMERATED_CURVE_ELEMENT_PURPOSE(.Z_Z_BENDING.)),(ENU-
MERATED_CURVE_ELEMENT_PURPOSE(.TORSION.)),(ENUMERATED_CURVE_ELEMENT_PUR-
POSE(.X_Y_SHEAR.),ENUMERATED_CURVE_ELEMENT_PURPOSE(.X_Z_SHEAR.))));
#608= CURVE_3D_ELEMENT_LOCATION_POINT_VARIABLE_VAL-
UES(#594,#215,.F.,(#609,#614),CURVE_VECTOR_2D_VARIABLE(.CURVE_BENDING_MO-
MENT.));
#663= CURVE_3D_ELEMENT_LOCATION_POINT_VARIABLE_VAL-
UES(#655,#215,.F.,(#664,#666),CURVE_VECTOR_2D_VARIABLE(.CURVE_BENDING_MO-
MENT.));
#695= CURVE_3D_ELEMENT_LOCATION_POINT_VARIABLE_VAL-
UES(#687,#215,.F.,(#696,#698),CURVE_VECTOR_2D_VARIABLE(.CURVE_BENDING_MO-
MENT.));
#182= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.1','NASTRAN
PBARL',(#183),(#195,#197),(#199,#202));
#221= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.2','NASTRAN
PBARL',(#222),(#227,#229),(#231,#233));
#244= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.3','NASTRAN
PBARL',(#245),(#250,#252),(#254,#256));
#267= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.4','NASTRAN
PBARL',(#268),(#273,#275),(#277,#279));
#290= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.5','NASTRAN
PBARL',(#291),(#296,#298),(#300,#302));
#313= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.6','NASTRAN
PBARL',(#314),(#319,#321),(#323,#325));
#336= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.7','NASTRAN
PBARL',(#337),(#342,#344),(#346,#348));
#359= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.8','NASTRAN
PBARL',(#360),(#365,#367),(#369,#371));
#382= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.9','NASTRAN
PBARL',(#383),(#388,#390),(#392,#394));
#405= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.10','NASTRAN
PBARL',(#406),(#411,#413),(#415,#417));
#428= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.11','NASTRAN
PBARL',(#429),(#434,#436),(#438,#440));
#451= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.12','NASTRAN
PBARL',(#452),(#457,#459),(#461,#463));
```

```
#474= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.13','NASTRAN
PBARL',(#475),(#480,#482),(#484,#486));
#497= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.14','NASTRAN
PBARL',(#498),(#503,#505),(#507,#509));
#520= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.15','NASTRAN
PBARL',(#521),(#526,#528),(#530,#532));
#543= CURVE_3D_ELEMENT_PROPERTY('PBARL.1.16','NASTRAN
PBARL',(#544),(#549,#551),(#553,#555));
#210= CURVE_3D_ELEMENT_REPRESENTA-
TION('1',(#206),#9,(#58,#63),#32,#215,#182,#212);
#241= CURVE_3D_ELEMENT_REPRESENTA-
TION('2',(#237),#9,(#63,#67),#32,#215,#221,#212);
#264= CURVE_3D_ELEMENT_REPRESENTA-
TION('3',(#260),#9,(#67,#71),#32,#215,#244,#212);
#287= CURVE_3D_ELEMENT_REPRESENTA-
TION('4',(#283),#9,(#71,#75),#32,#215,#267,#212);
#310= CURVE_3D_ELEMENT_REPRESENTA-
TION('5',(#306),#9,(#75,#79),#32,#215,#290,#212);
#333= CURVE_3D_ELEMENT_REPRESENTA-
TION('6',(#329),#9,(#79,#83),#32,#215,#313,#212);
#356= CURVE_3D_ELEMENT_REPRESENTA-
TION('7',(#352),#9,(#83,#87),#32,#215,#336,#212);
#379= CURVE_3D_ELEMENT_REPRESENTA-
TION('8',(#375),#9,(#87,#91),#32,#215,#359,#212);
#402= CURVE_3D_ELEMENT_REPRESENTA-
TION('9',(#398),#9,(#91,#95),#32,#215,#382,#212);
#425= CURVE_3D_ELEMENT_REPRESENTA-
TION('10',(#421),#9,(#95,#99),#32,#215,#405,#212);
#448= CURVE_3D_ELEMENT_REPRESENTA-
TION('11',(#444),#9,(#99,#103),#32,#215,#428,#212);
#471= CURVE_3D_ELEMENT_REPRESENTA-
TION('12',(#467),#9,(#103,#107),#32,#215,#451,#212);
#494= CURVE_3D_ELEMENT_REPRESENTA-
TION('13',(#490),#9,(#107,#111),#32,#215,#474,#212);
#517= CURVE_3D_ELEMENT_REPRESENTA-
TION('14',(#513),#9,(#111,#115),#32,#215,#497,#212);
#540= CURVE_3D_ELEMENT_REPRESENTA-
TION('15',(#536),#9,(#115,#119),#32,#215,#520,#212);
#563= CURVE_3D_ELEMENT_REPRESENTA-
TION('16',(#559),#9,(#119,#123),#32,#215,#543,#212);
#609= CURVE_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),#611,$);
#614= CURVE_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),#615,$);
#664= CURVE_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),#611,$);
#666= CURVE_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),#615,$);
#696= CURVE_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),#611,$);
#698= CURVE_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),#615,$);
#195= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#197= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#227= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#229= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#250= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#252= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#273= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
```

```
#275= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#296= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#298= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#319= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#321= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#342= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#344= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#365= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#367= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#388= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#390= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#411= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#413= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#434= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#436= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#457= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#459= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#480= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#482= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#503= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#505= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#526= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#528= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#549= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#551= CURVE_ELEMENT_END_OFFSET(#36,(0.,0.,0.));
#199= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#202= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#231= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#233= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#254= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#256= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#277= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#279= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#300= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#302= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#323= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#325= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#346= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#348= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#369= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#371= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#392= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#394= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#415= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#417= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#438= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#440= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#461= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#463= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#484= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#486= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#507= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#509= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#530= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#532= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#553= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#555= CURVE_ELEMENT_END_RELEASE(#36,(#200));
#200= CURVE_ELEMENT_END_RELEASE_PACKET(ENUMERATED_CURVE_ELEMENT_FREE-
DOM(.NONE.),0.);
```

```
#183= CURVE_ELEMENT_INTERVAL_CONSTANT(#184,#192,#163);
#222= CURVE_ELEMENT_INTERVAL_CONSTANT(#223,#224,#163);
#245= CURVE_ELEMENT_INTERVAL_CONSTANT(#246,#247,#163);
#268= CURVE_ELEMENT_INTERVAL_CONSTANT(#269,#270,#163);
#291= CURVE_ELEMENT_INTERVAL_CONSTANT(#292,#293,#163);
#314= CURVE_ELEMENT_INTERVAL_CONSTANT(#315,#316,#163);
#337= CURVE_ELEMENT_INTERVAL_CONSTANT(#338,#339,#163);
#360= CURVE_ELEMENT_INTERVAL_CONSTANT(#361,#362,#163);
#383= CURVE_ELEMENT_INTERVAL_CONSTANT(#384,#385,#163);
#406= CURVE_ELEMENT_INTERVAL_CONSTANT(#407,#408,#163);
#429= CURVE_ELEMENT_INTERVAL_CONSTANT(#430,#431,#163);
#452= CURVE_ELEMENT_INTERVAL_CONSTANT(#453,#454,#163);
#475= CURVE_ELEMENT_INTERVAL_CONSTANT(#476,#477,#163);
#498= CURVE_ELEMENT_INTERVAL_CONSTANT(#499,#500,#163);
#521= CURVE_ELEMENT_INTERVAL_CONSTANT(#522,#523,#163);
#544= CURVE_ELEMENT_INTERVAL_CONSTANT(#545,#546,#163);
#184= CURVE_ELEMENT_LOCATION(#185);
#223= CURVE_ELEMENT_LOCATION(#185);
#246= CURVE_ELEMENT_LOCATION(#185);
#269= CURVE_ELEMENT_LOCATION(#185);
#292= CURVE_ELEMENT_LOCATION(#185);
#315= CURVE_ELEMENT_LOCATION(#185);
#338= CURVE_ELEMENT_LOCATION(#185);
#361= CURVE_ELEMENT_LOCATION(#185);
#384= CURVE_ELEMENT_LOCATION(#185);
#407= CURVE_ELEMENT_LOCATION(#185);
#430= CURVE_ELEMENT_LOCATION(#185);
#453= CURVE_ELEMENT_LOCATION(#185);
#476= CURVE_ELEMENT_LOCATION(#185);
#499= CURVE_ELEMENT_LOCATION(#185);
#522= CURVE_ELEMENT_LOCATION(#185);
#545= CURVE_ELEMENT_LOCATION(#185);
#611= CURVE_ELEMENT_LOCATION(#612);
#615= CURVE_ELEMENT_LOCATION(#616);
#163= CURVE_ELEMENT_SECTION_DERIVED_DEFINITIONS('PBARL.1',0.,8.,(CONTEXT_DE-
PENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),(10.6666666666667,2.66666666666667,0.),7.32416666666667,CON-
TEXT_DEPENDENT_MEASURE(0.),(CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPEND-
ENT_MEASURE(0.)),(CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),(CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),CON-
TEXT_DEPENDENT_MEASURE(0.1),CONTEXT_DEPENDENT_MEASURE(13.3333333333333));
#136= DATA_ENVIRONMENT('DATA_ENV','Property conditions',(#132));
#148= DESCRIPTION_ATTRIBUTE('Elasticity',#147);
#154= DESCRIPTION_ATTRIBUTE('MassDensity',#153);
#160= DESCRIPTION_ATTRIBUTE('TangentCTE',#159);
#37= DIRECTION('refX',(1.,0.,0.));
#39= DIRECTION('AxisZ',(0.,0.,1.));
#45= DIRECTION('refX',(1.,0.,0.));
#47= DIRECTION('AxisZ',(0.,0.,1.));
#174= DIRECTION('',(1.,0.));
#208= DIRECTION('dxy',(0.,7.54979E-8,1.));
#239= DIRECTION('dxy',(0.,7.54979E-8,1.));
#262= DIRECTION('dxy',(0.,7.54979E-8,1.));
#285= DIRECTION('dxy',(0.,7.54979E-8,1.));
#308= DIRECTION('dxy',(0.,7.54979E-8,1.));
#331= DIRECTION('dxy',(0.,7.54979E-8,1.));
#354= DIRECTION('dxy',(0.,7.54979E-8,1.));
#377= DIRECTION('dxy',(0.,7.54979E-8,1.));
#400= DIRECTION('dxy',(0.,7.54979E-8,1.));
```

```
#423= DIRECTION('dxy',(0.,7.54979E-8,1.));
#446= DIRECTION('dxy',(0.,7.54979E-8,1.));
#469= DIRECTION('dxy',(0.,7.54979E-8,1.));
#492= DIRECTION('dxy',(0.,7.54979E-8,1.));
#515= DIRECTION('dxy',(0.,7.54979E-8,1.));
#538= DIRECTION('dxy',(0.,7.54979E-8,1.));
#561= DIRECTION('dxy',(0.,7.54979E-8,1.));
#212= ELEMENT_MATERIAL('1','Fea Material',(#147,#153,#159));
#603= ELEMENT_NODAL_FREEDOM_ACTIONS(#594,#575,(#604));
#658= ELEMENT_NODAL_FREEDOM_ACTIONS(#655,#575,(#659));
#690= ELEMENT_NODAL_FREEDOM_ACTIONS(#687,#575,(#691));
#604= ELEMENT_NODAL_FREEDOM_TERMS(#44,#601,(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UN-
SPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECI-
FIED_VALUE(.UNSPECIFIED.)));
#659= ELEMENT_NODAL_FREEDOM_TERMS(#44,#601,(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UN-
SPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECI-
FIED_VALUE(.UNSPECIFIED.)));
#691= ELEMENT_NODAL_FREEDOM_TERMS(#44,#601,(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UN-
SPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECI-
FIED_VALUE(.UNSPECIFIED.)));
#192= EULER_ANGLES((0.,0.,0.));
#224= EULER_ANGLES((0.,0.,0.));
#247= EULER_ANGLES((0.,0.,0.));
#270= EULER_ANGLES((0.,0.,0.));
#293= EULER_ANGLES((0.,0.,0.));
#316= EULER_ANGLES((0.,0.,0.));
#339= EULER_ANGLES((0.,0.,0.));
#362= EULER_ANGLES((0.,0.,0.));
#385= EULER_ANGLES((0.,0.,0.));
#408= EULER_ANGLES((0.,0.,0.));
#431= EULER_ANGLES((0.,0.,0.));
#454= EULER_ANGLES((0.,0.,0.));
#477= EULER_ANGLES((0.,0.,0.));
#500= EULER_ANGLES((0.,0.,0.));
#523= EULER_ANGLES((0.,0.,0.));
#546= EULER_ANGLES((0.,0.,0.));
#36= FEA_AXIS2_PLACEMENT_3D('CSYS.0',#41,#39,#37,.CARTESIAN.,'');
#44= FEA_AXIS2_PLACEMENT_3D('CSYS.1',#49,#47,#45,.CARTESIAN.,'');
#169= FEA_CURVE_SECTION_GEOMETRIC_RELATIONSHIP(#163,#170);
#144= FEA_LINEAR_ELASTICITY('',FEA_ISOTROPIC_SYMMETRIC_TEN-
SOR4_3D((10000000.,0.33)));
#151= FEA_MASS_DENSITY('',0.000254);
#147= FEA_MATERIAL_PROPERTY_REPRESENTATION(#146,#149,#136);
#153= FEA_MATERIAL_PROPERTY_REPRESENTATION(#152,#155,#136);
#159= FEA_MATERIAL_PROPERTY_REPRESENTATION(#158,#161,#136);
#32= FEA_MODEL_3D('FEM analysis',(#36),#6,'',(''),'');
#31= FEA_MODEL_DEFINITION('','',#30,.F.);
#185= FEA_PARAMETRIC_POINT('PBARL.1.fpp',(1.,0.,0.));
#612= FEA_PARAMETRIC_POINT('force_1',(-1.,0.,0.));
#616= FEA_PARAMETRIC_POINT('force_2',(1.,0.,0.));
#157= FEA_TANGENTIAL_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION('',ISO-
TROPIC_SYMMETRIC_TENSOR2_3D(1.3E-5));
```

```
#576= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),ENUMER-
ATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),ENUMERATED_DEGREE_OF_FREE-
DOM(.Z_TRANSLATION.),ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),ENUMER-
ATED_DEGREE_OF_FREEDOM(.Y_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTA-
TION.)));
#589= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),ENUMER-
ATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),ENUMERATED_DEGREE_OF_FREE-
DOM(.Z_TRANSLATION.)));
#596= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),ENUMER-
ATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),ENUMERATED_DEGREE_OF_FREE-
DOM(.Z_TRANSLATION.),ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),ENUMER-
ATED_DEGREE_OF_FREEDOM(.Y_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTA-
TION.)));
#601= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),ENUMER-
ATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),ENUMERATED_DEGREE_OF_FREE-
DOM(.Z_TRANSLATION.),ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),ENUMER-
ATED_DEGREE_OF_FREEDOM(.Y_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTA-
TION.)));
#579= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLA-
TION.),CONTEXT_DEPENDENT_MEASURE(1.));
#582= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLA-
TION.),CONTEXT_DEPENDENT_MEASURE(1.));
#583= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLA-
TION.),CONTEXT_DEPENDENT_MEASURE(1.));
#584= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTA-
TION.),CONTEXT_DEPENDENT_MEASURE(1.));
#585= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.Y_ROTA-
TION.),CONTEXT_DEPENDENT_MEASURE(1.));
#586= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTA-
TION.),CONTEXT_DEPENDENT_MEASURE(1.));
#6= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */(GEO-
METRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CON-
TEXT((#1,#2,#3,#4,#5))REPRESENTATION_CONTEXT('GLOBAL FEA CONTEXT','3d'));
#51= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */(GE-
OMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CON-
TEXT((#1,#2,#3,#4,#5))REPRESENTATION_CONTEXT('CSYS.1','3d'));
#178= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT
*/(GEOMETRIC_REPRESENTATION_CONTEXT(2)GLOBAL_UNIT_ASSIGNED_CONTEXT((#1))REP-
RESENTATION_CONTEXT('2d-planar',''));
#9= /* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT
*/(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CON-
TEXT()REPRESENTATION_CONTEXT('default parametric context','3D'));
#24= IDENTIFICATION_ROLE('default-role',$);
#133= ID_ATTRIBUTE('1',#131);
#127= ITEM_DEFINED_TRANSFORMATION('',$,#44,#36);
#1= /* LENGTH_UNIT+SI_UNIT */(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT(.MILLI.,.ME-
TRE.));
#627= LINEARLY_SUPERIMPOSED_STATE('22','LOAD');
#631= LINEARLY_SUPERIMPOSED_STATE('LOAD.22','load collector');
#674= LINEARLY_SUPERIMPOSED_STATE('23','LOAD');
#677= LINEARLY_SUPERIMPOSED_STATE('LOAD.23','load collector');
#4= /* MASS_UNIT+SI_UNIT */(MASS_UNIT()NAMED_UNIT(*)SI_UNIT(.KILO.,.GRAM.));
#146= MATERIAL_PROPERTY('1.0',$,#140);
#152= MATERIAL_PROPERTY('1.1',$,#140);
#158= MATERIAL_PROPERTY('1.2',$,#140);
#129= MEASURE_REPRESENTATION_ITEM('thermodynamic temperature unit',THERMODY-
NAMIC_TEMPERATURE_MEASURE(70.),#5);
```

```
#591= NODAL_FREEDOM_ACTION_DEFINITION(#588,#123,#36,#589,(CONTEXT_DEPEND-
ENT_MEASURE(-1000.),CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),.APPLIED_LOADS.);
#636= NODAL_FREEDOM_ACTION_DEFINITION(#635,#99,#36,#589,(CONTEXT_DEPEND-
ENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),.APPLIED_LOADS.);
#638= NODAL_FREEDOM_ACTION_DEFINITION(#635,#123,#36,#589,(CONTEXT_DEPEND-
ENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),.APPLIED_LOADS.);
#643= NODAL_FREEDOM_ACTION_DEFINITION(#642,#103,#36,#589,(CONTEXT_DEPEND-
ENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),.APPLIED_LOADS.);
#645= NODAL_FREEDOM_ACTION_DEFINITION(#642,#107,#36,#589,(CONTEXT_DEPEND-
ENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),.APPLIED_LOADS.);
#647= NODAL_FREEDOM_ACTION_DEFINITION(#642,#111,#36,#589,(CONTEXT_DEPEND-
ENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),.APPLIED_LOADS.);
#649= NODAL_FREEDOM_ACTION_DEFINITION(#642,#115,#36,#589,(CONTEXT_DEPEND-
ENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),.APPLIED_LOADS.);
#651= NODAL_FREEDOM_ACTION_DEFINITION(#642,#119,#36,#589,(CONTEXT_DEPEND-
ENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),CONTEXT_DEPENDENT_MEAS-
URE(0.)),.APPLIED_LOADS.);
#595= NODAL_FREEDOM_VALUES(#594,#599,#36,#596,(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UN-
SPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECI-
FIED_VALUE(.UNSPECIFIED.)));
#656= NODAL_FREEDOM_VALUES(#655,#599,#36,#596,(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UN-
SPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECI-
FIED_VALUE(.UNSPECIFIED.)));
#688= NODAL_FREEDOM_VALUES(#687,#599,#36,#596,(UNSPECIFIED_VALUE(.UNSPECI-
FIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UN-
SPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECI-
FIED_VALUE(.UNSPECIFIED.)));
#599=
NODE_GROUP('','',#32,(#58,#63,#67,#71,#75,#79,#83,#87,#91,#95,#99,#103,#107,
#111,#115,#119,#123));
#55= NODE_SET('node_set
1',(#58,#63,#67,#71,#75,#79,#83,#87,#91,#95,#99,#103,#107,#111,#115,#119,#12
3));
#58= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('1',(#59,#44),#51,#32);
#63= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('2',(#64,#44),#51,#32);
#67= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('3',(#68,#44),#51,#32);
#71= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('4',(#72,#44),#51,#32);
#75= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('5',(#76,#44),#51,#32);
#79= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('6',(#80,#44),#51,#32);
#83= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('7',(#84,#44),#51,#32);
#87= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('8',(#88,#44),#51,#32);
#91= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('9',(#92,#44),#51,#32);
#95= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('10',(#96,#44),#51,#32);
#99= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('11',(#100,#44),#51,#32);
#103= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('12',(#104,#44),#51,#32);
#107= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('13',(#108,#44),#51,#32);
#111= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('14',(#112,#44),#51,#32);
#115= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('15',(#116,#44),#51,#32);
#119= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('16',(#120,#44),#51,#32);
#123= NODE_WITH_SOLUTION_COORDINATE_SYSTEM('17',(#124,#44),#51,#32);
```

```
#26= ORGANIZATION('JOTNE','JOTNE','');
#27= ORGANIZATION_ROLE('');
#594= OUTPUT_REQUEST_STATE('STEP 1 output request','',(#570));
#655= OUTPUT_REQUEST_STATE('STEP 2 output request','',(#621));
#687= OUTPUT_REQUEST_STATE('STEP 3 output request','',(#670));
#207= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#208);
#238= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#239);
#261= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#262);
#284= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#285);
#307= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#308);
#330= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#331);
#353= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#354);
#376= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#377);
#399= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#400);
#422= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#423);
#445= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#446);
#468= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#469);
#491= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#492);
#514= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#515);
#537= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#538);
#560= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#561);
#206= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#207);
#237= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#238);
#260= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#261);
#283= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#284);
#306= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#307);
#329= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#330);
#352= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#353);
#375= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#376);
#398= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#399);
#421= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#422);
#444= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#445);
#467= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#468);
#490= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#491);
#513= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#514);
#536= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#537);
#559= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#560);
#188= PARAMETRIC_REPRESENTATION_CONTEXT('FEA parametric context','Paramet-
ric');
#2= /* PLANE_ANGLE_UNIT+SI_UNIT */(NAMED_UNIT(*)PLANE_AN-
GLE_UNIT()SI_UNIT($,.RADIAN.));
#54= POINT_REPRESENTATION('point_rep 1',(#44,#55),#51);
#17= PRODUCT('/NULL','NASTRAN analysis','',(#14));
#138= PRODUCT('1','1','stock material',(#14));
#14= PRODUCT_CONTEXT('/NULL',#11,'');
#19= PRODUCT_DEFINITION('','',#18,#16);
#140= PRODUCT_DEFINITION('1','',#139,#16);
#16= PRODUCT_DEFINITION_CONTEXT('/NULL',#11,'analysis');
#18= PRODUCT_DEFINITION_FORMATION('default-id','',#17);
#139= PRODUCT_DEFINITION_FORMATION('1',$,#138);
#30= PRODUCT_DEFINITION_SHAPE('mesh shape',$,#19);
#21= PRODUCT_RELATED_PRODUCT_CATEGORY('linear_static_analysis','',(#17));
#142= PRODUCT_RELATED_PRODUCT_CATEGORY('filament_assembly',$,(#138));
#131= PROPERTY_DEFINITION('Material Property Definition',$,#130);
#132= PROPERTY_DEFINITION_REPRESENTATION(#131,#134);
#171= RECTANGULAR_AREA($,#173,2.,4.);
#134= REPRESENTATION('1',(#129),#6);
#149= REPRESENTATION('1.0',(#144),#6);
#155= REPRESENTATION('1.1',(#151),#6);
```

```
#161= REPRESENTATION('1.2',(#157),#6);
#187= REPRESENTATION('dpr0',(#185,#612,#616),#188);
#128= REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION('','',#54,#32,#127);
#56= SHAPE_DEFINITION_REPRESENTATION(#30,#54);
#172= SHAPE_REPRESENTATION('rectangular-section-representa-
tion',(#171),#178);
#575= SINGLE_POINT_CONSTRAINT_ELE-
MENT('SPC1.100.1',(#570,#621,#670),#58,#44,(#579,#582,#583,#584,#585,#586),'
');
#574= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#573,#575,#576,(CONTEXT_DEPEND-
ENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEAS-
URE(0.),CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),CON-
TEXT_DEPENDENT_MEASURE(0.)));
#3= /* SI_UNIT+SOLID_ANGLE_UNIT */(NAMED_UNIT(*)SI_UNIT($,.STERA-
DIAN.)SOLID_ANGLE_UNIT());
#5= /* SI_UNIT+THERMODYNAMIC_TEMPERATURE_UNIT */(NAMED_UNIT(*)SI_UNIT($,.DE-
GREE_CELSIUS.)THERMODYNAMIC_TEMPERATURE_UNIT());
#569= SPECIFIED_STATE('default input state','default input state');
#572= SPECIFIED_STATE('Base state 1','Base state');
#573= SPECIFIED_STATE('100','SPC1');
#588= SPECIFIED_STATE('200','FORCE');
#623= SPECIFIED_STATE('Base state 2','Base state');
#624= SPECIFIED_STATE('10','SPCADD');
#635= SPECIFIED_STATE('300','FORCE');
#642= SPECIFIED_STATE('400','FORCE');
#672= SPECIFIED_STATE('Base state 3','Base state');
#629= STATE_COMPONENT('LOAD.22','overall scale factor',#627,1.);
#633= STATE_COMPONENT('scaled LOAD.22.300','component scale fac-
tor',#631,1.);
#641= STATE_COMPONENT('scaled LOAD.22.400','component scale fac-
tor',#631,1.);
#675= STATE_COMPONENT('LOAD.23','overall scale factor',#674,1.);
#679= STATE_COMPONENT('scaled LOAD.23.200','component scale fac-
tor',#677,1.);
#682= STATE_COMPONENT('scaled LOAD.23.300','component scale fac-
tor',#677,1.);
#684= STATE_COMPONENT('scaled LOAD.23.400','component scale fac-
tor',#677,1.);
#587= STATE_RELATIONSHIP('','SPC relationship',#572,#573);
#593= STATE_RELATIONSHIP('','LOAD relationship',#572,#588);
#625= STATE_RELATIONSHIP('','SPC relationship',#624,#573);
#626= STATE_RELATIONSHIP('','SPC relationship',#623,#624);
#632= STATE_RELATIONSHIP('LOAD.22','load collector',#629,#631);
#640= STATE_RELATIONSHIP('','LOAD relationship',#633,#635);
#653= STATE_RELATIONSHIP('','LOAD relationship',#641,#642);
#654= STATE_RELATIONSHIP('','LOAD relationship',#623,#627);
#673= STATE_RELATIONSHIP('','SPC relationship',#672,#624);
#678= STATE_RELATIONSHIP('LOAD.23','load collector',#675,#677);
#681= STATE_RELATIONSHIP('','LOAD relationship',#679,#588);
#683= STATE_RELATIONSHIP('','LOAD relationship',#682,#635);
#685= STATE_RELATIONSHIP('','LOAD relationship',#684,#642);
#686= STATE_RELATIONSHIP('','LOAD relationship',#672,#674);
#33= STRUCTURAL_RESPONSE_PROPERTY('','',#31);
#34= STRUCTURAL_RESPONSE_PROPERTY_DEFINITION_REPRESENTATION(#33,#32);
ENDSEC;

END-ISO-10303-21;
```