STEP AP209 ed2
Linear Static Structural FEA Handbook

Volume 1: FEA Input for LOTAR EAS Pilot Study #1

Release 2.2
May 16, 2018

Contacts:

CAE-IF		
Jochen Boy PROSTEP AG Dolivostraße 11 64293 Darmstadt / Germany jochen.boy@prostep.com	Jean-Marc Crepel AFNeT 30, Rue de Miromesnil 75008 Paris / France jean-marc.crepel@afnet.fr	Phil Rosché ACCR, LLC. 125 King Charles Circle Summerville, SC 29485 USA phil.rosche@accr-llc.com

LOTAR EAS		
Chris Johnson Technical LOTAR EAS WG - Americas Lockheed Martin john.c.johnson@lmco.com	Joseph G. Draper Co-leader LOTAR EAS WG – Americas The Boeing Company joseph.g.draper@boeing.com	Albert Lévy Co-leader LOTAR EAS WG - Europe CIMPA on behalf of Airbus albert.levy.external@airbus.com

Table of Contents

List of Figures

Trademark Acknowledgments

NASTRAN is a registered trademark of NASA.

VERSION LOG

Version Description	By	Revision	Date
Original Draft	JCJ/KH	1.8	8/30/2017
Added Material Product Figure, Updated Figure References and Introduction	JCJ	1.9	3/22/2018
Accept all changes and release for review Add links to EAS material in References and file links	JCJ JCJ/AL	2.0 2.1	4/10/2018 4/16/2018
Update title and cover sheet with CAE-IF logo and contacts	JCJ	2.2	5/16/2018

JCJ - John C Johnson
AL - Albert Levy
JD - Joe Draper
KH - Keith Hunten

1 Introduction

This document is a handbook describing the finite element models (FEM) used for the LOTAR International Engineering Analysis and Simulation (EAS) pilot studies. These pilot studies focused on development of converter implementations of the International Standards Organization (ISO) 10303-209 ed2 (AP209 ed2) Standard, one Part of the 10303 family of Standards commonly known as STEP (Standard for The Exchange of Product data). The implementations were subsequently used to generate AP209 ed2 STEP files used as input to the initial test round of the Computer Aided Engineering Implementer Forum (CAE_IF) .

This handbook can also be used as training material for an introduction to the AP209 ed2 data model for finite element models. The document requires at least a cursory understanding of the Finite Element Method, finite element models and the information technologies involved in applying the AP209 ed2 application protocol. Familiarity with the details of the EXPRESS data modeling language and fundamental concepts of the STEP standard parts are not necessarily a prerequisite for using this document, but will greatly improve the readers understanding. The handbook can also help the reader focus on those parts of STEP needed to encode FEM information and the documentation resources available to producers, consumers and implementers of this information.

2 Pilot Study Overview

The pilot study was created to encourage software providers to participate in an initial effort to develop commercial tools capable of converting traditional finite element model information, expressed in solver ASCII format, to the AP209 ed2 format. A second goal of the pilot study was to produce a reference set of AP209 ed2 STEP files that have been validated with the AP209 ed2 EXPRESS schema and have been checked for semantics with respect to the Recommended Practices for AP 209 ed2 10303-209:2014 and the document, Geometric Founding and Associativity in ISO 10303-209, Rev. B, 2/15/2001.

The pilot study provides a small test suite of FEA models to be translated from a native solver format, in this case NASTRAN, to the STEP AP209 ed2 ASCII file representation (Part 21). These models are prismatic beam models with various linear element types, boundary conditions, load cases, and output requests. The overall geometry for each model is the same and is simple (see Figure 1). This simplicity was intentional to focus the effort on the interpretation of the FEM definitions and constructs, not the geometric abstractions related to meshing techniques.

Implied Model Units:

- Length: inch
- Force: lbf
- Mass: lbf-s^2/in

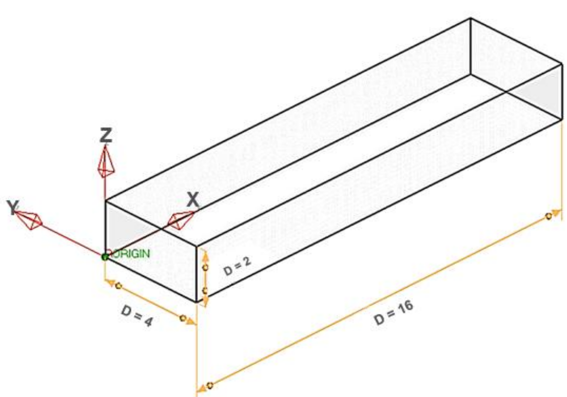


Figure 1 Pilot Model Geometry

Discretization of the geometry to the finite element model domain was accomplished through the abstractions and resulting finite element models shown in Figure 2. The finite element models were exported in NASTRAN bulk data format for use in this pilot study.

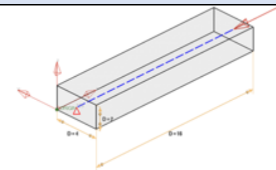
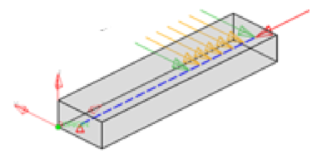
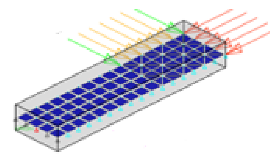
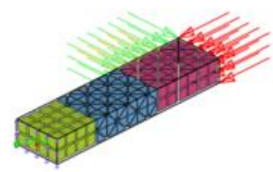
Abstraction - Geometry	NASTRAN Element Type	Pilot Model
1 Dimensional - Line	Linear CROD [Axial Stiffness – 1 DOF]	
1 Dimensional - Line	Linear CBAR [Bending/Axial/Shear – 5 DOF]	
2 Dimensional - Surface	Linear CQUAD4/CTRIA3 [Bending/Membrane/Shear – 5 DOF]	
3 Dimensional - Solid	Linear CHEXA/CPENTA/CTETRA [Bending/Membrane/Shear – 3 DOF]	

Figure 2 Pilot Model Abstractions

The boundary conditions of the pilot models were established so that there is no constraint to Poisson's expansion or contraction. This modeling of the boundary conditions was used to model correctly the classical solution of the bar in compression and bending. This is critical to having a 'correct' solution easily available to check the analysis results.

The first load case for all of the pilot models is a simple axial compression case where the compression loads are kinematically consistent. The deflection and stress of the beam are calculated by the following formula:

Stress = Load/Cross-sectional area

Strain = Stress/Modulus of Elasticity

Axial Deflection = Strain*Length

In the remainder of this document each pilot model is discussed in detail, where complete listings of the solver input and the STEP output are included in the Appendices.

3 AP209 Overview

This section introduces several of the information technologies used in working with the STEP standard, and in particular those standards used via AP209 ed2. The reader can use this introduction to locate and explore in detail the topics highlighted herein. The links provided in Section 9 References and Links are a good starting place for additional reading.

AP209, formally known as ISO-10303-209 ed2, Application Protocol: Multidisciplinary Analysis and Design, provides the data structures for the exchange of part and configuration identification with configuration control data, with or without associated 3D part model information. AP209 was developed under the auspices of the International Organization for Standardization (ISO), Technical Committee 184, Sub-Committee 4, and is one of a series of parts comprising the full STandard for the Exchange of Product model data (STEP) standard known as ISO 10303.

3.1 STEP Module and Resource Library

Application Protocols such as AP209 ed2 are published in HTML format. All STEP Application Protocols are built from common modules in the SMRL (STEP Module and Resource Library) thus assuring integration and interoperability. The HTML format provides a convenient web browser format to locate definitions for all the entities, types and rules needed for this pilot study, along with the associated EXPRESS schemas. These documents are copyrighted and published by ISO, so therefore the reader is referred to the ISO web site for access information.

The SMRL contains all the 10303 STEP standards other than the Application Protocols themselves, and consists of the following 10303 Parts:

- Integrated resources series of parts: ISO 10303 Parts 41 to 112;
- Application modules series of parts: ISO 10303 Parts 401 to 499 and 1001 to 1999;
- Application interpreted constructs series of parts: ISO 10303 Parts 501 to 599;
- Logical model of expressions: ISO 13584 Part 20;
- Business object models series of parts: ISO 10303 Parts 3001 to 3099.

The initial architecture of ISO 10303 did not use the Application Modules in SMRL. That architecture was more monolithic and resulted in consistency issues as the standards grew in scope. The Application Modules SMRL architecture was introduced to solve these scalability issues.

As shown in Figure 3, AP209 ed2 is a modular STEP standard focused on multidisciplinary analysis and simulation information, along with related design (primarily shape) and PDM/PLM information. As such, it is built from smaller building blocks such as application modules which are used to form a multi-layered set of schema definitions.

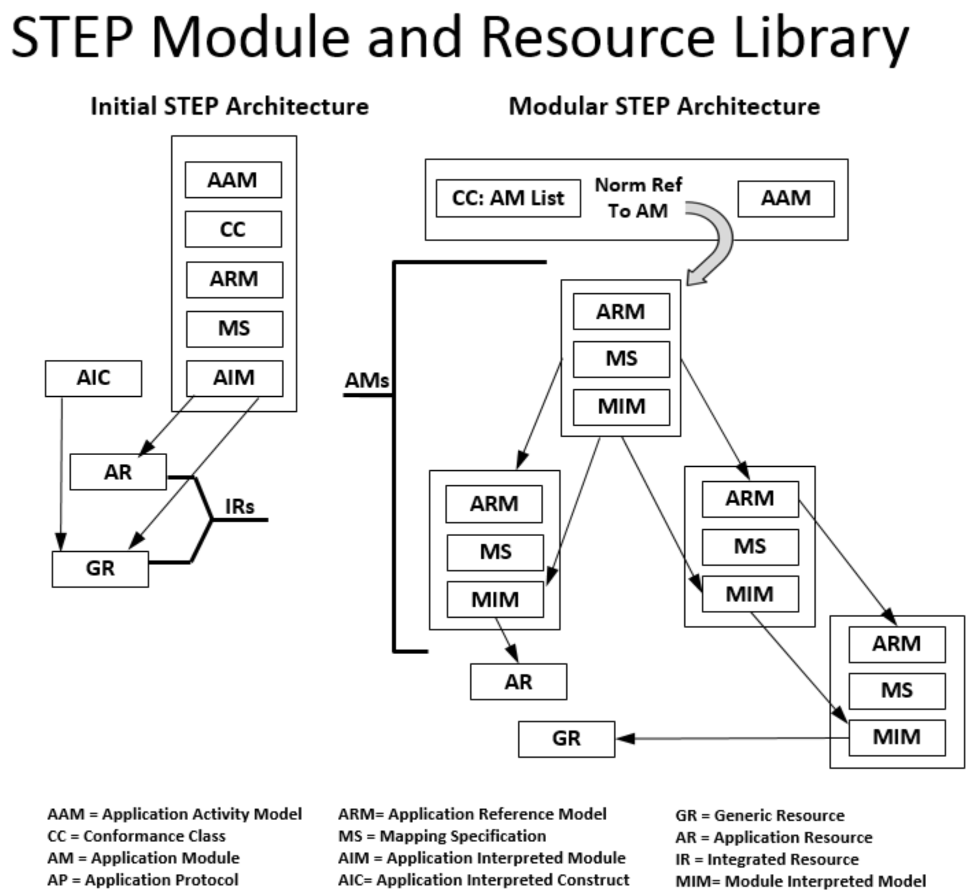


Figure 3 Evolution of the STEP Architecture and Current SMRL

These building blocks are documented in the SMRL Part documents. The schemas in these building blocks are defined using the EXPRESS schema modeling language (Part 10303-11). In this document, EXPRESS schema fragments are presented that relate to the specific constructs used for each pilot model.

A fundamental concept of the STEP architecture is the use of various higher level data models that are mapped to lower level data models. This concept is illustrated in Figure 3 and in the AP209 ed2 Recommended Practice discussion of how standards support the user community needs. This is accomplished by documenting the flow of the everyday engineering actions put forth in the AP209 ed2 Application Activity Model (AAM) using data elements defined in the Application Reference Model (ARM). The ARM documents these information requirements in terms of the application experts vernacular. The ARM is in turn tied by mappings to the constructs in the Integrated Resources to create the interpreted Application Model (AIM). The AIM EXPRESS model is used to implement the physical data sharing, typically by using the implementation format specified by the 10303-21 clear text (ASCII) encoding.

The ARM and AIM data model schemas are related to each other through mapping specifications that are accessible in the SMRL. There is no discussion of the mapping specifications in this document. The schema fragments used in this document are all AIM schema definitions.

There are two views of the AP209 ed2 ARM and AIM schemas that are available: a short form and a long form. A short form schema includes all the building blocks through the recursive 'USE FROM' EXPRESS construct that references other modular building blocks. The long form view is the result of resolving all the references and concatenating the resulting schema constructs into a single schema file. Both are available in the SMRL. The short form of AP209 ed2 is shown in Figure 4. Note that the first line includes the AP 242 standard in its entirety. Also note that only the AIM long form EXPRESS schema is used for implementations.

```
SCHEMA Ap209_multidisciplinary_analysis_and_design_mim;
    USE FROM Ap242_managed_model_based_3d_engineering_mim;   -- ISO/TS 10303-442
    USE FROM Analysis_assignment_mim;   -- ISO/TS 10303-1474
    USE FROM Analysis_characterized_mim;   -- ISO/TS 10303-1475
    USE FROM Analysis_identification_mim;   -- ISO/TS 10303-1476
    USE FROM Analysis_product_relationships_mim;   -- ISO/TS 10303-1373
    USE FROM Analysis_mim;   -- ISO/TS 10303-1372
    USE FROM Analysis_representation_mim;   -- ISO/TS 10303-1785
    USE FROM Cfd_model_mim;   -- ISO/TS 10303-1377
    USE FROM Finite_elements_mim;   -- ISO/TS 10303-1383
    USE FROM Mesh_connectivity_mim;   -- ISO/TS 10303-1384
    USE FROM Mesh_topology_mim;   -- ISO/TS 10303-1386
    USE FROM Product_analysis_mim;   -- ISO/TS 10303-1387
    USE FROM Analysis_shape_mim;   -- ISO/TS 10303-1374
    USE FROM Finite_element_shape_mim;   -- ISO/TS 10303-1382
    USE FROM Cfd_specified_general_property_mim;   -- ISO/TS 10303-1379
    USE FROM Fea_definition_relationships_mim;   -- ISO/TS 10303-1380
    USE FROM Fea_material_aspects_mim;   -- ISO/TS 10303-1769
    USE FROM Cfd_equations_mim;   -- ISO/TS 10303-1376
    USE FROM Cfd_results_mim;   -- ISO/TS 10303-1378
    USE FROM Finite_element_analysis_mim;   -- ISO/TS 10303-1381
    USE FROM Mesh_function_mim;   -- ISO/TS 10303-1385
    USE FROM mathematical_functions_schema;   -- ISO 10303-50
    USE FROM iso13584_generic_expressions_schema;   -- ISO 13584-20
    USE FROM iso13584_expressions_schema;   -- ISO 13584-20
    USE FROM Part_view_definition_mim;   -- ISO/TS 10303-1023
    USE FROM Collection_identification_and_version_mim;   -- ISO/TS 10303-1396
    TYPE ap209e2_analysed_item <details removed for clarity> END_TYPE;
    TYPE ap209e2_analysis_approval_item < details removed for clarity> END_TYPE;
    TYPE ap209e2_analysis_identification_item < details removed for clarity> END_TYPE;
    TYPE ap209e2_groupable_item < details removed for clarity>  END_TYPE;
END_SCHEMA;  -- Ap209_multidisciplinary_analysis_and_design_mim
```

Figure 4 ISO 10303 AP209 ed2 Short Form Schema

3.2 External Physical File – Part 21

As noted above, the typical way to externalize an AP209 ed2 data model instance is via a physical file that is an ASCII encoding of the data based on the EXPRESS constructs in AP209 ed2 mapped through ISO 10303-21 (see Figure 5).

```
/*************************************************************************
* STEP Physical File Fragment
*************************************************************************/
#35= FEA_AXIS2_PLACEMENT_3D('0',#33,#31,#29,.CARTESIAN.,'FEA_BASIC_COORD_SYSTEM.0');
#29= DIRECTION('RefX',(1.,0.,0.));
#31= DIRECTION('AxisZ',(0.,0.,1.));
#33= CARTESIAN_POINT('',(0.,0.,0.));
```

Figure 5 STEP Physical File Fragment (Part 21)

This physical file format is called a Part 21 STEP file. The reader is referred to the ISO 10303-21 document for complete syntax rules. Basic syntax rules will be introduced as needed in the context of the pilot model being discussed. The pilot models in this document are all encoded as Part 21 STEP files. Validation of the contents of these STEP files is accomplished by reading the Part 21 files into tools that can assess adherence to the underlying schema definitions and/or semantics.

4 AP209 ed2 Data Model Organization

Due to the large scope of the Standard, the AP209 ed2 schema is large and complex. Many views of the data model are presented in the application protocol (AP209 ed2) and recommended practices (RP) documents. In addition to the figures in the RP, Figure 6 shown below illustrates refinements of the RP information to show how the data model can be considered as smaller conceptual topics and how those topics are related to the pilot models.

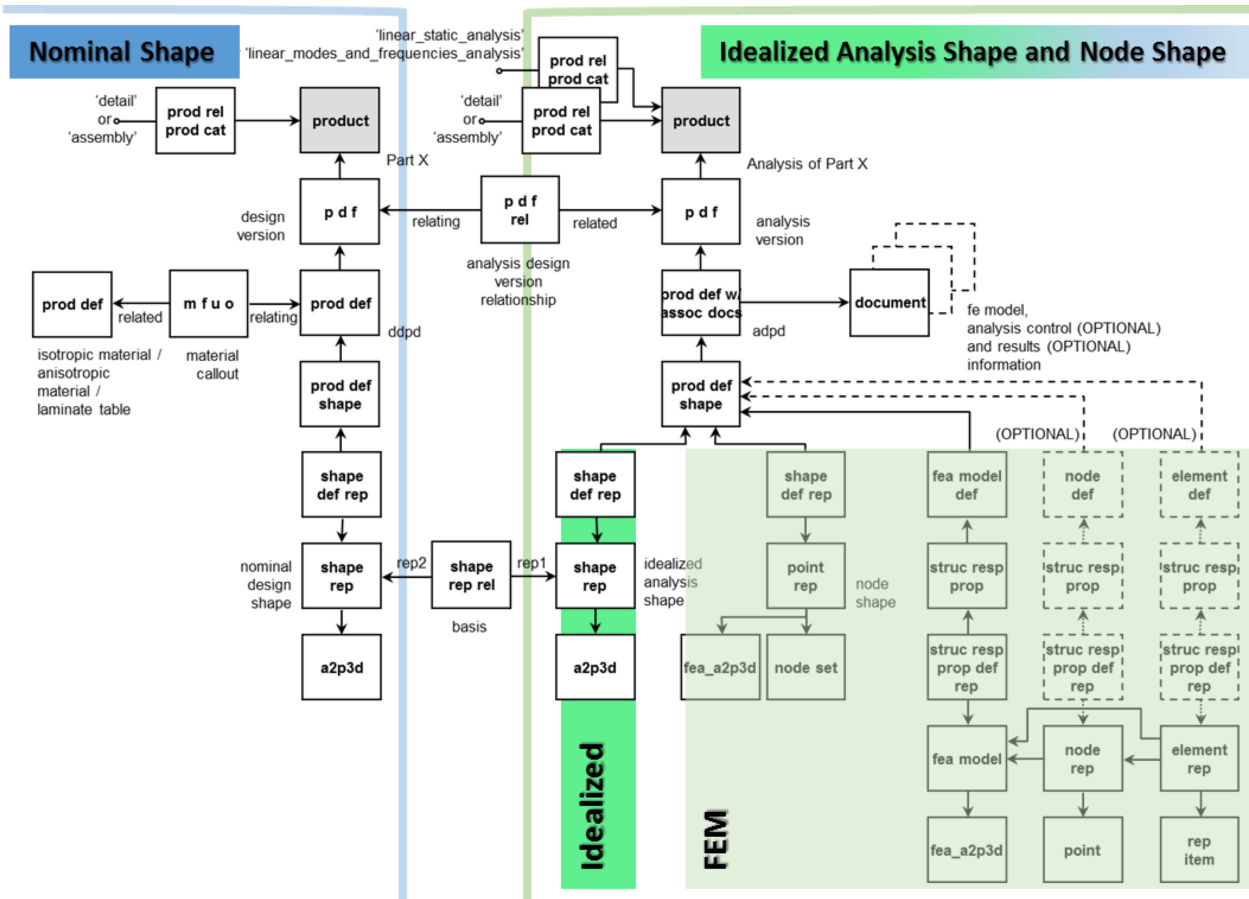


Figure 6 AP209 ed2 Relationship Between Design and Analysis

This view clearly illustrates how the design geometry (nominal shape) is related to the abstracted analysis geometry (idealized analysis shape) and the finite element model (node shape).

The pilot models in this Handbook use a native solver format (NASTRAN) as the FEM information source and therefore no nominal design or idealized analysis geometry is captured. However, the tools being developed under this pilot study will eventually be required to include the AP209 ed2 constructs that are used to capture the full FEA model definition and solution results. Full FEA model definition includes associativity to nominal design geometry or the idealized analysis geometry used for mesh generation and for loads and boundary condition specifications.

Traditionally, FEA solver models do not specify a system of units but are required to be internally consistent. Recent practices have begun to define explicitly a consistent set of units as part of the solver input for FEA model definition. The STEP standard can capture this information when constructing model contexts and associated unit entities. The pilot models do not explicitly specify a system of units, however, there is an implied use of the English inch-pound force-second system due to the existence of recognized material property values. While it may be possible to deduce a likely system of units for a given input file, it is recommended that direct user input define explicitly which system of units applies. How to specify this information for the conversion process is an active area of development and discussion.

4.1 Analysis Product Metadata

The right side of Figure 6 represents the AP209 ed2 analysis product. The figure hides many entities that deal with metadata about the product for the sake of clarity. Figure 7 includes more of these entities and categorizes the metadata of the analysis product; this figure was created from an early STEP file generated for the pilot study using the 1st pilot model.

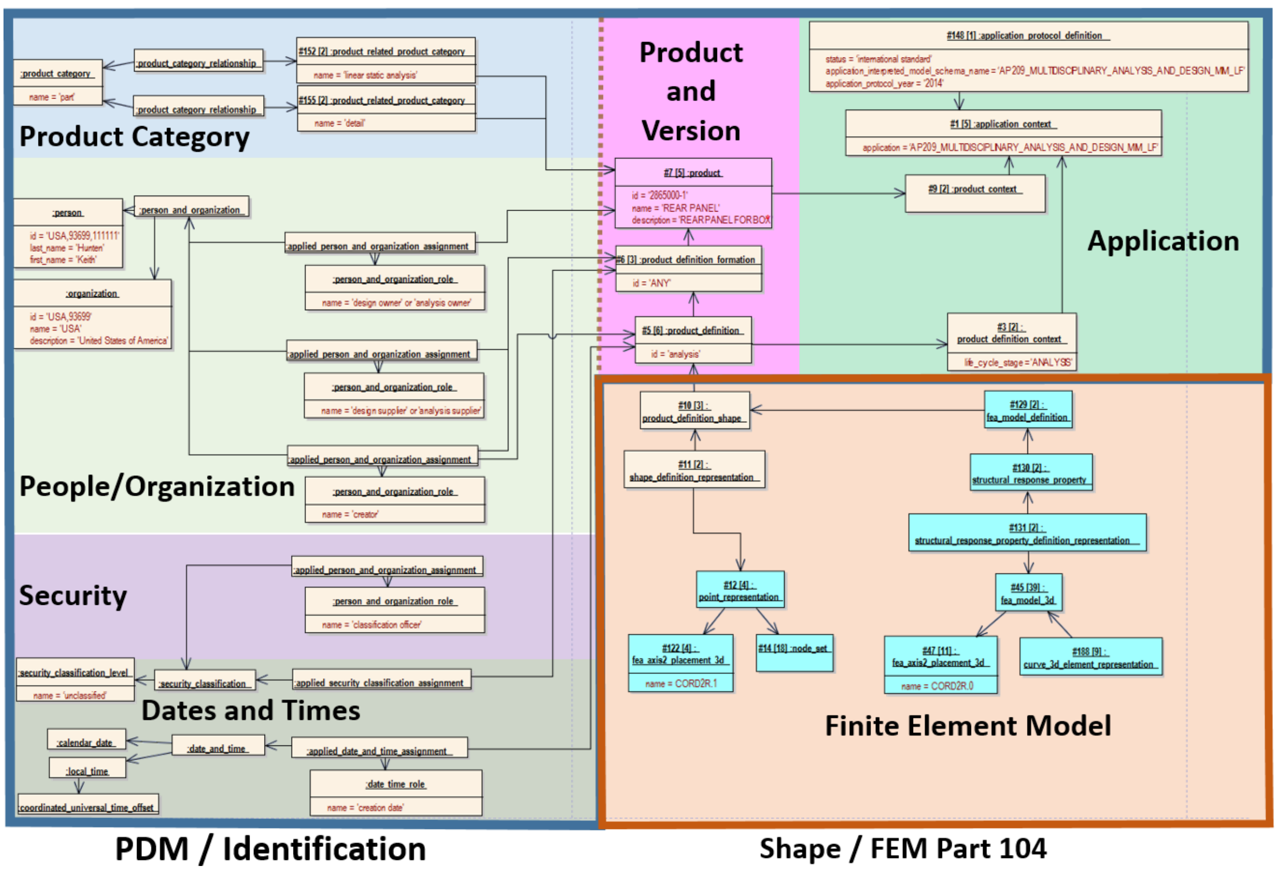


Figure 7 AP209 ed2 Analysis Product Categories

The ‘PDM/Identification’ area of Figure 7 represents much of the metadata that is used to understand the pedigree of the analysis product. Metadata such as who authored it, who owns it, who reviewed and approved it, security and retention statements and important dates can be defined here. None of this information will be directly available from the NASTRAN solver decks used as the information source for the pilot project. However, the metadata can either be supplied by the conversion process or defaulted to a minimal set sufficient for further workflow processing.

An important characteristic to note about Figure 7 is that all the instances of ‘PDM/Identification’ metadata point inward to the entities contained in the ‘Product and Version’ area. This implies that from an EXPRESS entity definition point of view, all the ‘PDM/Identification’ entities are optional. While some commercial CAD tools offer options to specify these entities in the tool’s graphical user interface (GUI), if the option is not selected, the AP242 STEP file written will not contain this information. This produces a well-formed STEP file, but it is not semantically correct and complete.

This information may appear optional by looking only at the referencing direction, but the EXPRESS definition may also include rules which require existence of these entities. Additionally, the AP209 ed2 Recommended Practices documentation may dictate that these entities should exist and may provide enumerations of acceptable attribute values.

The ‘Product Category’ area of Figure 7 illustrates this requirement. The EXPRESS schema fragment from the AP209 ed2 long form for the Product_Category entity is shown in Figure 8. These entities are defined in the ISO 10303-41 Integrated generic resource: Fundamentals of product description and support (Part 41).

```
ENTITY Product_Category;
      name          : Label;
      description   : OPTIONAL Text;
    DERIVE
      id  : Identifier := Get_Id_Value( SELF );
    WHERE
      wr1: ( SIZEOF( USEDIN( SELF, 'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' +
ID_ATTRIBUTE.IDENTIFIED_ITEM' ) ) <= 1 );
END_ENTITY;

ENTITY Product_Category_Relationship;
      name           : Label;
      description    : OPTIONAL Text;
      category       : Product_Category;
      sub_category   : Product_Category;
    WHERE
      wr1: Acyclic_Product_Category_Relationship( SELF, [SELF.sub_category] );
END_ENTITY;

ENTITY Product_Related_Product_Category
    SUBTYPE OF ( Product_Category );
      products   : SET [1 : ?] OF Product;
END_ENTITY;
```

Figure 8 EXPRESS Schema for Product Category and Related Entities

The relationship object enables a hierarchy of product category information to be specified for any product. The Product_Category super-type only requires the ‘name’ attribute be present while the Product_Related_Product_Category sub-type requires a reference to at least one Product. In addition to these structural requirements, the AP209 ed2 Recommended Practice states:

- [Section 2.8.2.2] – "It is strongly recommended that all implementations of AP 209 ed2 establish an instance of **product_category** with a **name** attribute value of `'part'`."
- [Section 2.11.1.1] – "AP 209 ed2 requires that all **product**s exist in at least one **product_related_product_category**. This restriction (**product_requires_product_category**) forces all parts into one of the following categories: `'detail'`, `'assembly'`, `'inseparable assembly'`, or `'customer furnished equipment'`."
- [Section 2.11.1.1] – "An analysis **product** should also be assigned to a **product_related_product_category** that indicates the type of analysis. The appropriate categories in AP 209 ed2 are: `'linear static analysis'` and `'linear modes and frequencies analysis'`."

The EXPRESS schema and recommended practice declarations translate to a structure that at a minimum should look like the instance diagram shown in Figure 9 with the 'name' attribute being populated as shown.

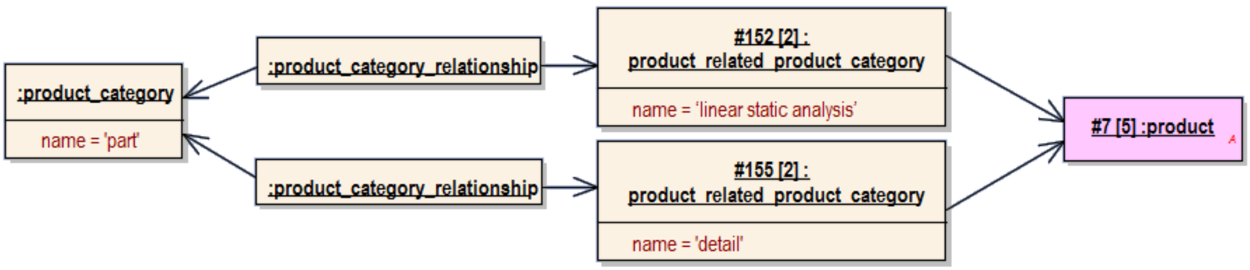


Figure 9 Product Category Instance Diagram

For the pilot study, a minimal set of metadata should be included but is not the primary focus. It is discussed in this handbook to introduce how the implementation requirements are expressed in the schemas and recommended practice, and the inherent problem with using native solver information as the source for the 'PDM/Identification' metadata.

4.2 Application Protocol and Analysis Product Identification

Identification of the data model and the analysis product itself are accomplished using the same methods as a typical part design product. The application protocol definition and application context define the schema used to construct the data model, whereas the product and related product definition entities provide for identification and independent versioning of the analysis product. Again, the recommended practice document provides guidance as to how these entities should be populated. To summarize:

- The **application_context** entity identifies the application that defines the data. The **application** attribute, based on its definition in ISO 10303-41, should have the value `'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF'`
- The **application_protocol_definition** entity further identifies the AP. For AP 209 ed2, the **status** attribute, based on its definition in ISO 10303-41, should have the value `'international standard'`. The **application_interpreted_model_schema_name** attribute should have the value `'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF'` and the **application_protocol_year** attribute should have the value `'2014'`
- The **product** should contain identifying information about the part being analyzed

- The **product_definition_formation** should contain a unique identifier or the string '**ANY'** if it is a generic un-versioned part or analysis
- The **product_definition** should have an identifier populated with the string '**analysis'**

All these constructs are included in AP209 ed2 through the EXPRESS 'USE FROM' statement and are defined in ISO 10303-41 Integrated generic resource: Fundamentals of product description and support (Part 41). As noted above the USE FROM technique enforces the common use of the data constructs in the Integrated Resources amongst all Application Protocols providing interoperability.

Unlike metadata discussed in the prior section of this handbook, these elements of the AP209 ed2 standard are not optional and must be included for a STEP file to be valid. Figure 10 illustrates a sample instance diagram.

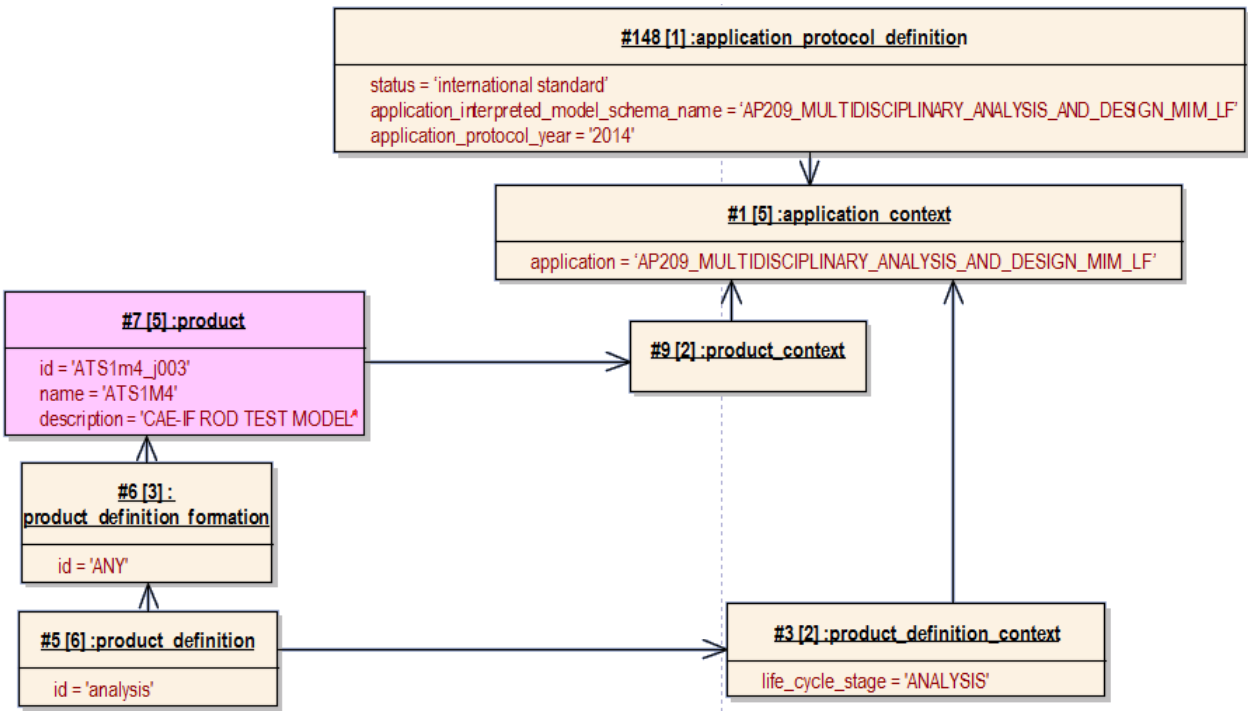


Figure 10 Example Analysis Protocol and Product Identification

There are other aspects of the product information that can be expressed with the AP209 ed2 standard such as the details of all the possible captured metadata, but that is not the focus of this document or the pilot study. The preceding discussion is only an introduction to the topics and the reader is encouraged to refer to the actual standard documents and recommended practices for complete coverage.

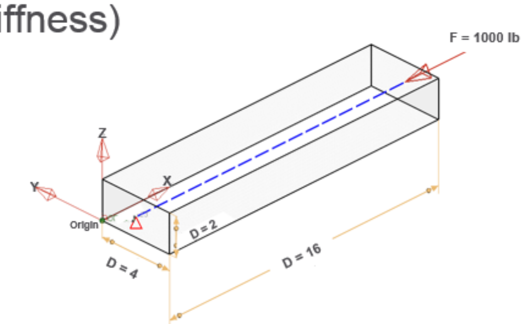
The remainder of this document focuses on the lower right corner of Figure 7; it captures the definitions of the finite element pilot models. The contents of this part of the data model are documented in detail in the ISO 10303-104 Integrated application resource: Finite element analysis document (Part 104). Part 104 and the associated schema definitions are the primary sources of information needed to map the native solver information (NASTRAN) found in the pilot project input files, to the AP209 ed2 data models and produce the familiar Part 21 STEP file representation.

Each pilot model is discussed in a similar format. For each topic, such as grid points, an instance diagram is shown to illustrate how the information appears in the AP209 ed2 data model, followed by the source NASTRAN information, and then the STEP file entities listed along with the related AP209 ed2 schema fragments.

5 Pilot Model ATS1 – 1D Rod Element

The ATS1 pilot model composition is illustrated in Figure 11. While simplistic, this model is sufficient to demonstrate a basic understanding of the AP209 ed2 data model requirements.

- Axial stiffness element (no torsional stiffness)
- FE model composition
  - Elements: 16 CROD
  - Nodes: 17 GRID
  - Loads: 1 FORCE
  - Boundary: 1 SPC1
  - Property: 1 PROD
  - Material: 1 MAT1 (aluminum)
  - System: 1 CORD2R (at origin)



Axial stress: 1000 / 8 = 125 psi
Axial strain: 125 / 10e+6 = 1.25e-5 in/in
Axial defl: 1.25e-5 x 16 = 0.0002 in

- Subcase and output requests
  - Subcases: 1 SUBCASE
  - Boundary: 1 SPC
  - Loads: 1 LOAD
  - Output: 4 GPFORCE (global), DISPLACEMENT, SPCFORCES, STRESS

*See listing for output parameters

Figure 11 Pilot Model Content Summary for ATS1

Additional details of the model are:

- Single Isotropic aluminum material property at 70 deg F.
- 1000 lb axial load in compressive (-x) direction
- Single rectangular coordinate system located at the origin (0,0,0)
- All nodes and elements are on a line defined as y=-2 and z=1 in the local system.

5.1 Product Definition Shape

The AP209 ed2 data model treats the finite element model (FEM) as an analysis product that follows a defined structure for STEP products. This was discussed in Section 4. That discussion is extended here to illustrate the path from the product entity to the primary entities used to capture the finite element model information. Figure 12 illustrates this structure, wherein the entity instances follow the naming convention, ***#id [number of references] : entity type***.

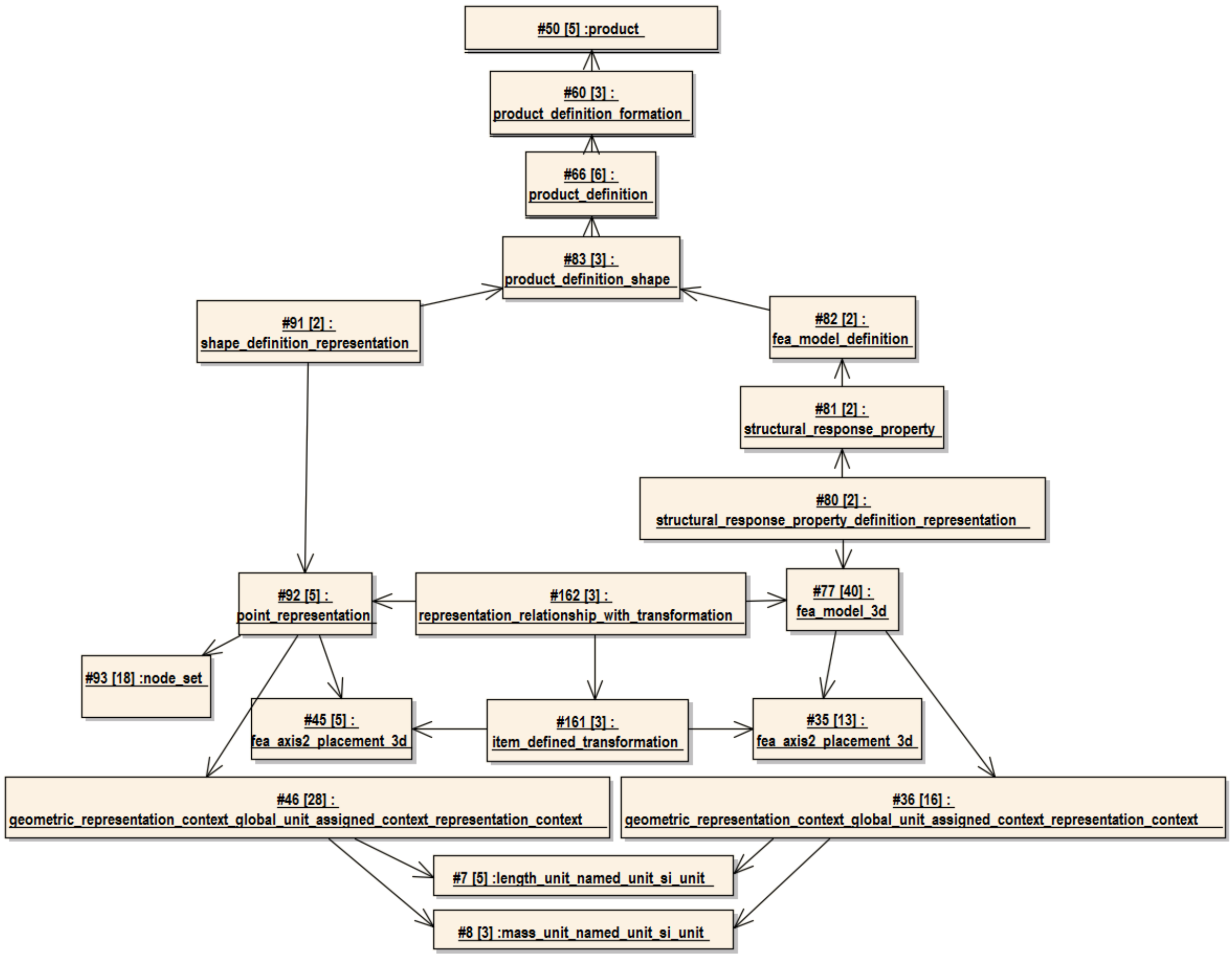


Figure 12 Overall Data Model Structure From Analysis Product to Contexts and Units

The product_definition_shape links the standard product identification and versioning to the details of the FE model. The two entities that refer to this entity link the representational aspects and definitional aspects of the finite element model, respectively. These are important topics but again, not the focus of this pilot study.

This handbook is concerned with the entities related to the point_represention and the fea_model_3d. The remainder of this discussion deals with information associated with these two entities.

Figure 12 shows the overall framework used to structure the details of this pilot model and illustrates the basic geometric founding aspects of AP209 ed2. A basic tenant of the STEP data model is that any schema entity in the representation.items list of a representation entity has an appropriate context (specified in representation.context_of_items) to describe how to interpret the information related to the representation. For geometric entities such as points, nodes and coordinate systems, a complex context is used to define the dimension of the geometric space; it can reference a set of units and can also define a global uncertainty tolerance. The use of the EXPRESS type context_dependent_measure which is simply a REAL value indicates that the using representation identifies the context of the measure. In this manner, the units associated with the measure can be ascertained. However, in some cases, the entity type measure_with_unit is specified directly in the EXPRESS schema. In this case, explicit instancing of a measure_value and a unit is required. Lastly, there is the concept of 'unspecified' that indicates there is no appropriate value for the attribute.

5.2 FEA Model

The fea_model_3d represents the existence of the FEA model and the implied (or asserted) basic coordinate frame of reference it is described in. There is no explicit input data that maps to this STEP entity; it is used for identification of the FEA model, what software may have been used to create it and the name of the intended analysis code. The Part 21 fragment for this pilot model is shown in Figure 13.

```
#77= FEA_MODEL_3D('Identification',(#35),#36,
'NASTRAN BDF Converter v0.0.0',('NASTRAN'),'AnalysisModelType');
```

Figure 13 STEP File Fragment for FEA Model 3D

The corresponding flattened EXPRESS schema entity definition for fea_model_3d is shown in Figure 14.

```
ENTITY fea_model_3d;
    ENTITY representation;
        name                                    : LABEL;
        items                                   : SET [1:?] OF representation_item;
        context_of_items                        : representation_context;
      DERIVE
        id                                      : IDENTIFIER;
        description                             : TEXT;
    ENTITY fea_model;
        creating_software                       : TEXT;
        intended_analysis_code                  : SET [1:?] OF TEXT;
        analysis_type                           : TEXT;
    ENTITY fea_model_3d;
END ENTITY;
```

Figure 14 Flattened EXPRESS Schema for FEA_Model_3D

A flattened EXPRESS schema definition shows the inheritance hierarchy and which super-type an attribute is defined in. Derived attributes do not appear in the Part 21 file explicitly.

In the Part 21 fragment, note that a specification of a SET of things (items and intended_analysis_code) result in the use of parenthesis around a comma separated list of items. In this case, each attribute is a list of only one entity. This syntactical rule also applies to a LIST of things, and appears often in the example model Part 21 file fragments. The list of nodes in the connectivity of an element follows this syntax and is shown in the element definition section

5.3 Coordinate Systems

The two complex contexts that are shown in the product_definition_shape diagram result from the basic frame of reference that the FEA model exists in (the basic coordinate system) and local frames of reference created by local coordinate systems. When multiple coordinate systems are used in a model such as this one, there must be a relationship that exists that can be used to transform information in one coordinate system to another. An FEA example of this would be transforming a node defined in a local coordinate system to the basic coordinate system of the FEM.

5.3.1 STEP Diagram

Figure 15 illustrates the coordinate transformation concept showing only the local instances related to the transformation.

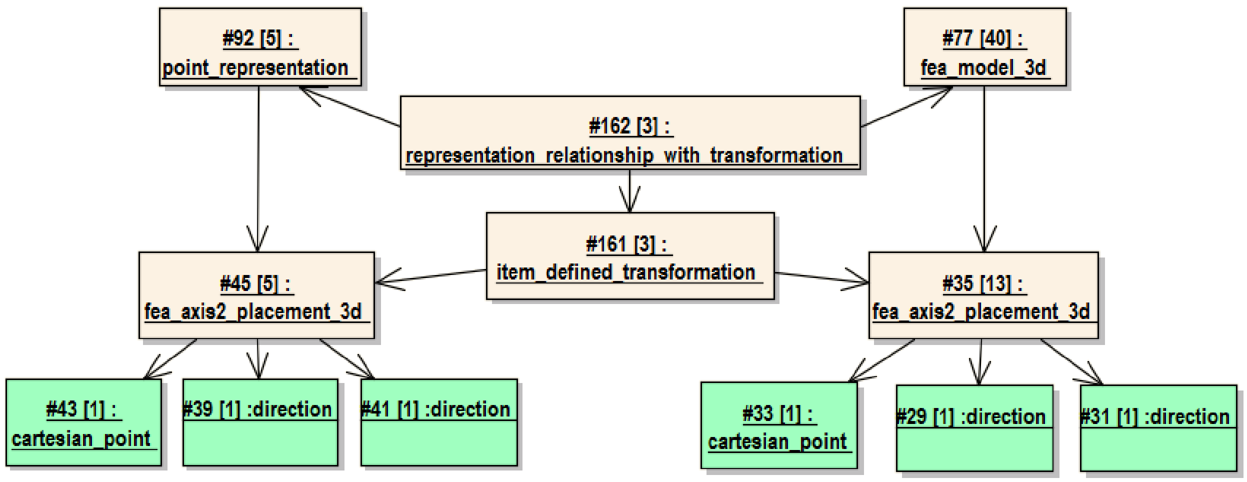


Figure 15 Details of item_defined_transformation Relating Coordinate Systems

An item_defined_transformation represents the fact that the two items it references are related to each other through a transformation operator that can be constructed from the information contained in the actual items being transformed. In this case, the item_defined_transformation relates the two fea_axis2_placement_3d (representation_items) entities to each other. It is also referenced by a representation_relationship_with_transformation that specifies which owning representations are to be used to access the related context information (such as system of units or uncertainty) for the item being transformed. This structure is convenient because the definition of the fea_axis2_placement_3d origin location and direction can be used to reconstruct the transformation operator without using trigonometric calculations that would be required if angular rotations were used to define the transformation.

5.3.2 FE Model Input

The NASTRAN specification of the local coordinate system and the input data schema is shown in Figure 16. The A, B and C points refer to the origin, a point on the Z axis and a point in the X-Z plane. In this case, the local coordinate system is coincident with the basic coordinate system.

1	2	3	4	5	6	7	8	9	10
CORD2R	CID	RID	A1	A2	A3	B1	B2	B3	
	C1	C2	C3						

```
CORD2R   1               0.       0.       0.       0.       0.       1.
         1.       0.     0.
```

Figure 16 NASTRAN Card Image for Rectangular Coordinate System

5.3.3 STEP Entities

The Part 21 instance data for the coordinate systems and transformation is shown in Figure 17. NASTRAN does not define the basic coordinate system for a model; it is implied. However, an explicit definition for the basic coordinate system is used in the AP209 ed2 data model. The axis2_placement_3d instance referenced in the set of items for the fea_model_3d instance, is the basic coordinate system.

```
#35= FEA_AXIS2_PLACEMENT_3D('0',#33,#31,#29,.CARTESIAN.,'FEA_BASIC_COORD_SYSTEM.0');

#36= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT('FEA_BASIC_COORD_SYSTEM.0','3d'));

#29= DIRECTION('RefX',(1.,0.,0.));
#31= DIRECTION('AxisZ',(0.,0.,1.));
#33= CARTESIAN_POINT('',(0.,0.,0.));

#45= FEA_AXIS2_PLACEMENT_3D('1',#43,#41,#39,.CARTESIAN.,'CORD2R.1');

#46= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT('CORD2R.1','3d'));

#39= DIRECTION('RefX',(1.,0.,0.));
#41= DIRECTION('AxisZ',(0.,0.,1.));
#43= CARTESIAN_POINT('',(0.,0.,0.));

#92= POINT_REPRESENTATION('point_representation_name_1',(#93,#45),#46);
#93= NODE_SET('node_set_name_1',(#90,#99,#103,#107,#111,#115,#119,#123,
#127,#131,#135,#139,#143,#147,#151,#155,#159));

#161= ITEM_DEFINED_TRANSFORMATION('','',#35,#45);;
#162= REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION('','',#77,#92,#161);
```

Figure 17 STEP File Fragment for Coordinate System Relationships

The node_set entity in the listing aggregates all the nodes that are using the local coordinate system for their geometric location identified by a point_representation. If additional coordinate systems are present, additional point_representations with fea_axis2_placement_3d and their associated node_sets will exist with the requisite item_defined_transformation. The EXPRESS schema entities for a fea_axis2_placement_3d are shown in Figure 18.

```
ENTITY fea_axis2_placement_3d;
    ENTITY representation_item;
        name                                        : LABEL;
    ENTITY geometric_representation_item;
      DERIVE
        dim                                         : DIMENSION_COUNT;
    ENTITY placement;
        location                                    : cartesian_point;
    ENTITY axis2_placement_3d;
        axis                                        : OPTIONAL direction;
        ref_direction                               : OPTIONAL direction;
      DERIVE
        p                                           : LIST [3:3] OF direction;
    ENTITY fea_axis2_placement_3d;
        system_type                                 : COORDINATE_SYSTEM_TYPE;
        description                                 : TEXT;
END_ENTITY;
```

Figure 18 EXPRESS Schema for STEP Coordinate System Definition

5.4 Node Definition

FEA models use grid points and a mesh of finite elements to discretize the continuous domain of the physical model into smaller regions that have numerical approximations that allow finite element solvers to produce calculated responses at these grid points. The cloud of grid points defines the basic shape of the finite element model. In AP209 ed2, grid points are mapped to node entities. The node entities in STEP use a reference to a point entity sub-type that defines the spatial location (coordinates) of the particular type of node.

NOTE: Do not confuse the title of this clause with the node_definition ENTITY in AP209 ed2.

5.4.1 STEP Diagram

The relationship of the AP209 ed2 node entity to the basic model and local coordinate system entities is shown in Figure 19.

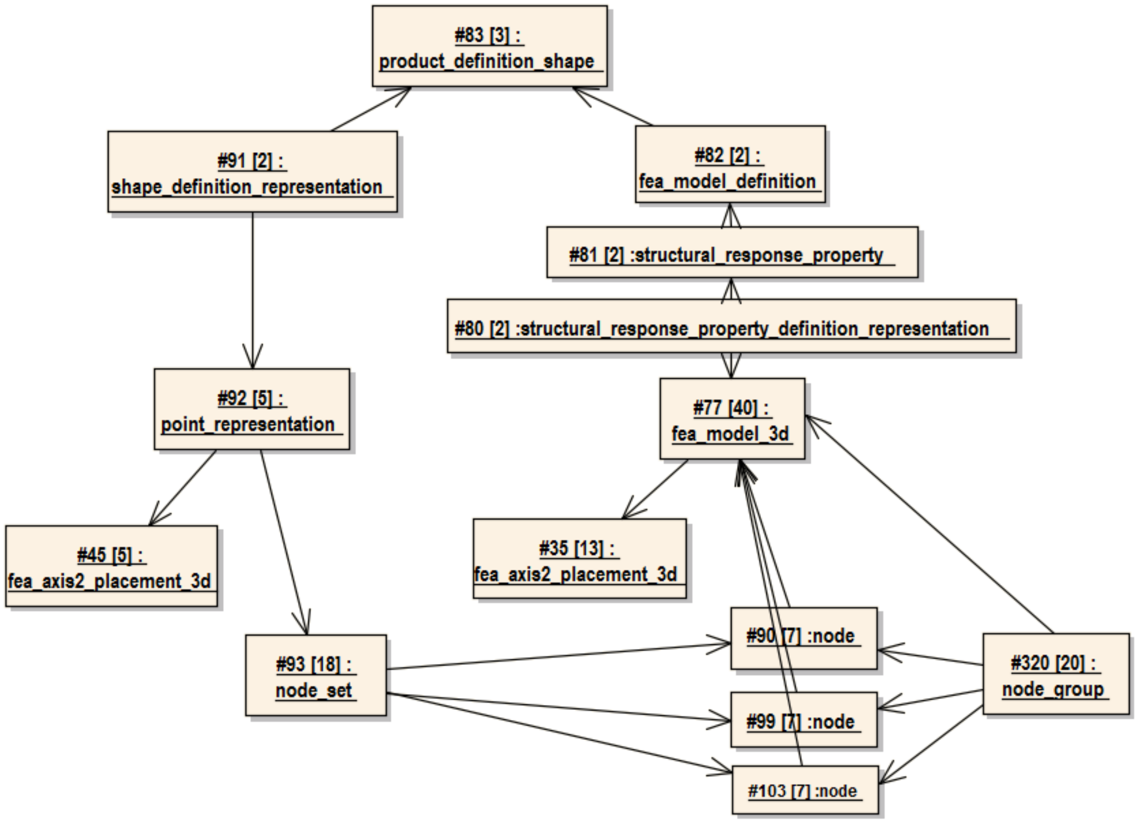


Figure 19 STEP Node Entity Relationships to other FEA Instances

The node instances all refer to the owning fea_model_3d and are referenced by other entities such as the node_set for geometric founding discussed in Section 5.1 and the node_group which represents the concept of a set of nodes from the solvers perspective. Do not confuse the node_set entity with a solver node 'set' or 'group' definition. The AP209 ed2 node_set is a STEP geometric founding construct related to coordinate system assignment.

Solver definitions of a 'set' or 'group' of grid points are used for model sub-domains, boundary conditions, loads, output requests and other solver dependent purposes. These are all mapped to instances of node_group.

5.4.2 FE Model Input

The pilot model consists of 17 equally spaced grid points with the first at x=0, y=-2, z=1 and the last at x=16, y=-2, z=1 in the local coordinate system. These grid points define the end points of the 1 dimensional rod element used in this model. The NASTRAN listing of the first 3 grid points is shown in Figure 20.

1	2	3	4	5	6	7	8	9	10
GRID	ID	CP	X1	X2	X3	CD	PS	SEID	

```
GRID    1       1       0.      -2.      1.
GRID    2       1       1.      -2.      1.
GRID    3       1       2.      -2.      1.
```

Figure 20 NASTRAN Card Image for Grid Points

The 'CP' field is a reference to the ID of the coordinate system that the X1, X2, X3 coordinate values should be interpreted in. The 'CD' field specifies the coordinate system that the degrees of freedom at this point are defined in as well as the reference system for output data at the grid point. The 'PS' is a permanent constraint applied to the degrees of freedom at this grid point and the 'SEID' is the super element identifier. These two additional fields are not relevant to this pilot model.

5.4.3 STEP Entities

The equivalent STEP listing for these 3 grid points is shown in Figure 21.

```
#90= NODE('1',(#88),#46,#77);
#88= CARTESIAN_POINT('1',(0.,-2.,1.));

#99= NODE('2',(#97),#46,#77);
#97= CARTESIAN_POINT('2',(1.,-2.,1.));

#103= NODE('3',(#101),#46,#77);
#101= CARTESIAN_POINT('3',(2.,-2.,1.));


#320= NODE_GROUP('','ALL',#77,(#90,#99,#103,#107,#111,#115,#119,#123,
#127,#131,#135,#139,#143,#147,#151,#155,#159));
#321= ID_ATTRIBUTE('ALL',#320);
```

Figure 21 STEP File Fragment for Nodes and Node Groups

Each node refers to a 3-dimensional context for geometric founding and a point entity sub-type to define the coordinates of the point. The label attribute is used to specify the node ID.

These entities can be located in Figure 19 for reference. The 'ALL' node_group resulted from an output request that indicated all nodal displacements or grid point forces be output. The EXPRESS flattened schema for these entities is shown in Figure 22.

```
ENTITY node;
    ENTITY representation;
        name                                         : LABEL;
        items                                        : SET [1:?] OF representation_item;
        context_of_items                             : representation_context;
      DERIVE
        id                                           : IDENTIFIER;
        description                                  : TEXT;
    ENTITY node_representation;
        model_ref                                    : fea_model;
    ENTITY node;
END_ENTITY;

ENTITY cartesian_point;
    ENTITY representation_item;
        name                                         : LABEL;
    ENTITY geometric_representation_item;
      DERIVE
        dim                                          : DIMENSION_COUNT;
    ENTITY point;
    ENTITY cartesian_point;
        coordinates                                  : LIST [1:3] OF LENGTH_MEASURE;
END_ENTITY;

ENTITY node_group;
    ENTITY group;
        name                                         : LABEL;
        description                                  : OPTIONAL TEXT;
      DERIVE
        id                                           : IDENTIFIER;
    ENTITY fea_group;
        model_ref                                    : fea_model;
    ENTITY node_group;
        nodes                                        : SET [1:?] OF node_representation;
END_ENTITY;

ENTITY node_set;
    ENTITY representation_item;
        name                                         : LABEL;
    ENTITY geometric_representation_item;
      DERIVE
        dim                                          : DIMENSION_COUNT;
    ENTITY node_set;
        nodes                                        : SET [1:?] OF node_representation;
END_ENTITY;
```

Figure 22 Flattened EXPRESS Schema for Node and Related Entities

The node entity inherits items attribute from the representation super-type which is a set of 1 or more representation_items. For a node entity, the set contains a single cartesian_point entity which defines the node location (a cartesian_point is a kind of representation_item). The context_of_items attribute contains the requisite reference to the context of the point entity. Additionally, the super-type node_representation adds the reference to the fea_model to a node entity. This pattern of context<-representation-> representation_item is a common theme throughout the STEP data model and the reader is encouraged to become familiar with this concept in the Part 43 reference document.

5.5 Element Definition

A simple linear rod element is defined between each grid point to supply stiffness to the grid points. Even though this is a simple 2 degree of stiffness element, it is a good illustration of how FEM elements are mapped to the STEP AP209 ed2 data model.

NOTE: Do not confuse the title of this clause with the element_definition ENTITY in AP209 ed2.

5.5.1 STEP Diagram

An overall view of the STEP implementation of element definition and relationships to the FEA model concepts such as element coordinate systems, physical properties and material properties is illustrated in Figure 23.

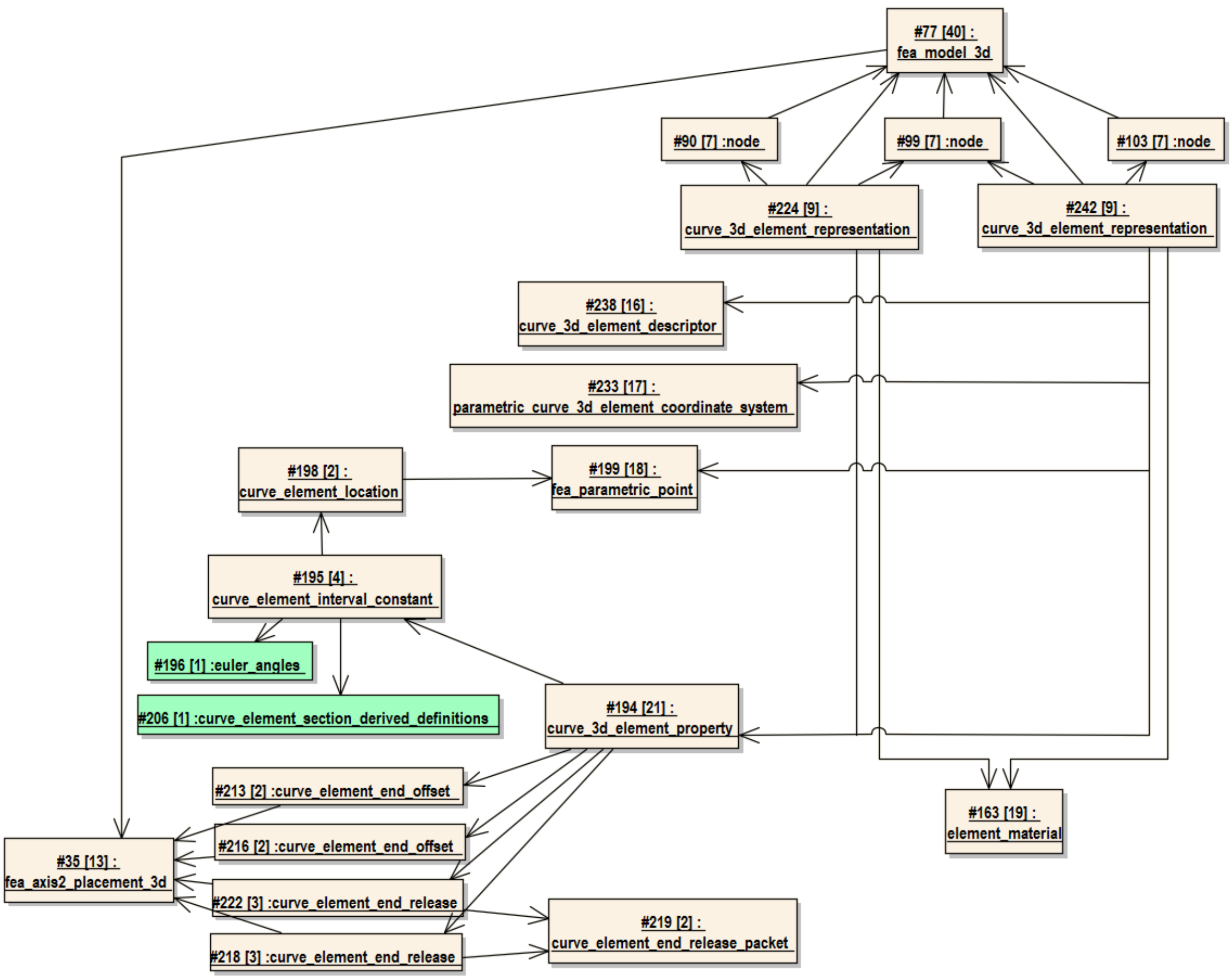


Figure 23 STEP Element Definition and Related Data Model Structures

This diagram starts with the fea_model_3d instance, shows the relevant nodes used in the items list for connectivity, and then the basic layout of the element properties. The element information is discussed in logical groupings below.

5.5.2 FE Model Input

There are 16 NASTRAN elements defined using the CROD input card image shown in Figure 24. The rod element used in this model supplies stiffness for only 2 degrees of freedom at each grid point.

1	2	3	4	5	6	7	8	9	10
CROD	EID	PID	G1	G2					

```
CROD     1       1       1       2
CROD     2       1       2       3
```

Figure 24 NASTRAN Card Image for Rod Element

Therefore, the element acts as a simple axial and torsional spring between the grid points. There are no additional options to be specified for this element.

5.5.3 STEP Entities

The 1-dimensional linear spring element maps to a curve_3d_element_representation. The behavioral aspects of the element are specified by the curve_3d_element_descriptor which is a set of enumerated capabilities for the element as shown in Figure 25.

```
#224= CURVE_3D_ELEMENT_REPRESENTATION('1',(#233,#199),#225,(#90,#99),#77,#238,#194,#163);
#242= CURVE_3D_ELEMENT_REPRESENTATION('2',(#233,#199),#225,(#99,#103),#77,#238,#194,#163);

#238= CURVE_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_CURVE.CROD',(
(ENUMERATED_CURVE_ELEMENT_PURPOSE(.AXIAL.)),
(ENUMERATED_CURVE_ELEMENT_PURPOSE(.TORSION.))
));
```

Figure 25 STEP File Fragment for Element and Element Descriptor

The flattened EXPRESS schema shown in Figure 26 defines, in order, what each attribute in the element instances represent.

```
ENTITY curve_3d_element_representation;
    ENTITY representation;
        name                                   : LABEL;
        items                                  : SET [1:?] OF representation_item;
        context_of_items                       : representation_context;
      DERIVE
        id                                     : IDENTIFIER;
        description                            : TEXT;
    ENTITY element_representation;
        node_list                              : LIST [1:?] OF node_representation;
    ENTITY curve_3d_element_representation;
        model_ref                              : fea_model_3d;
        element_descriptor                     : curve_3d_element_descriptor;
        property                               : curve_3d_element_property;
        material                               : element_material;
END_ENTITY;
```

Figure 26 EXPRESS Schema for Curve 3D Element Representation

The name attribute has been used to hold the solver element ID while the items set is a list of entity references used to establish a parametric element coordinate system. The context_of_items reference follows. Next, connectivity is specified by an ordered set of 2 node references. While the linear rod element node list order is straight forward, the higher dimension elements are more complicated as each solver has its own unique way of ordering the connectivity array.

The Part 104 document explicitly specifies how the node list should be ordered for all element types and orders. It is the translator's responsibility to map the solver order to the Part 104 order. This also includes supplying placeholder dummy nodes in the node list for a connectivity location that the solver may not support (such as mid-face nodes for solids in NASTRAN). The last four attributes are references to the owning fea_model_3d, the element descriptor, the physical properties and material properties.

The element descriptor specifies how the curve_3d_element_representation behavior is specialized to match the particular solver element type that this instance represents. In this case, a simple axial and torsional 1-dimensional spring element is the solver element behavior specified. The description field can be used to identify what type of element this descriptor represents. However, FEA solvers should not place any processing logic in this field unless agreed to and documented in the recommended practices.

As a note of caution, even though the solver element usage may allow for degrees of freedom to be released or a specific stiffness to be set to zero, the descriptor for that element should define the full capabilities of the element type and not the individual element options that are represented elsewhere in the STEP data model. This pilot model illustrates this issue by specifying a zero torsional stiffness property, however, the descriptor still includes the possible existence of torsional stiffness with the ENUMERATED_CURVE_ELEMENT_PURPOSE(.TORSION.) enumeration item.

The EXPRESS schema for the descriptor is shown in Figure 27.

```
ENTITY curve_3d_element_descriptor;
      ENTITY element_descriptor;
          topology_order                  : ELEMENT_ORDER;
          description                     : TEXT;
      ENTITY curve_3d_element_descriptor;
          purpose                         : SET [1:?] OF SET [1:?] OF CURVE_ELEMENT_PURPOSE;
END_ENTITY;

TYPE Curve_Element_Purpose = SELECT
  ( Enumerated_Curve_Element_Purpose,
   Application_Defined_Element_Purpose );
END_TYPE;

TYPE Enumerated_Curve_Element_Purpose = ENUMERATION OF
  ( Axial,
   Y_Y_Bending,
   Z_Z_Bending,
   Torsion,
   X_Y_Shear,
   X_Z_Shear,
   Warping );
END TYPE;
```

Figure 27 EXPRESS Schema for Curve 3D Element Descriptor

The first two references in the elements items list referred to entities needed to establish the parametric coordinate system for the element. Therefore, the context_of_items must relate to a parametric_context (or at least a parametric_context must be part of the full complex context for the element). These items are shown in Figure 28.

```
#233= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#236);
#236= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#234);
#234= DIRECTION('undefined',(1.,0.,0.));

#199= FEA_PARAMETRIC_POINT('',(1.,0.,0.));

#225= /* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CONTEXT()
REPRESENTATION_CONTEXT('REP_CONTEXT_FEA','3D'));
```

Figure 28 STEP File Fragment for Curve 3D Parametric Coordinate System

These instances define a parametric element coordinate system where the parametric coordinate ξ represents the position and direction along the length of the element from (-1,0,0) to (1,0,0) as shown in Figure 29.

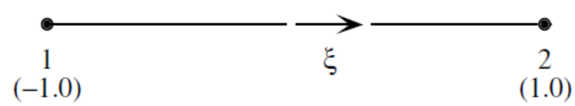


Figure 29 Curve Parametric Element Coordinates

The parametric point shown is used to define a parametric curve_element_location that defines the end of a curve_element_interval_constant where the physical properties are defined along the length of the overall element. This is discussed in the next section.

5.6 Property Definition

The property definition for curve elements can be complex. This simple model does not hide all the complexity but it simplifies the population of the required attributes. The pilot model uses a single NASTRAN property that is shared by all the rod elements. This is possible since there are no element unique attributes that must be captured in the STEP AP209 ed2 property representation. Examples of attributes that would require unique properties are offsets and releases or tapering that is specified uniquely on an element. The CROD element does not support any of these additional attributes.

5.6.1 STEP Diagram

The property section from the top level diagram shown in Figure 23 for this pilot model is repeated as Figure 30 for reference.

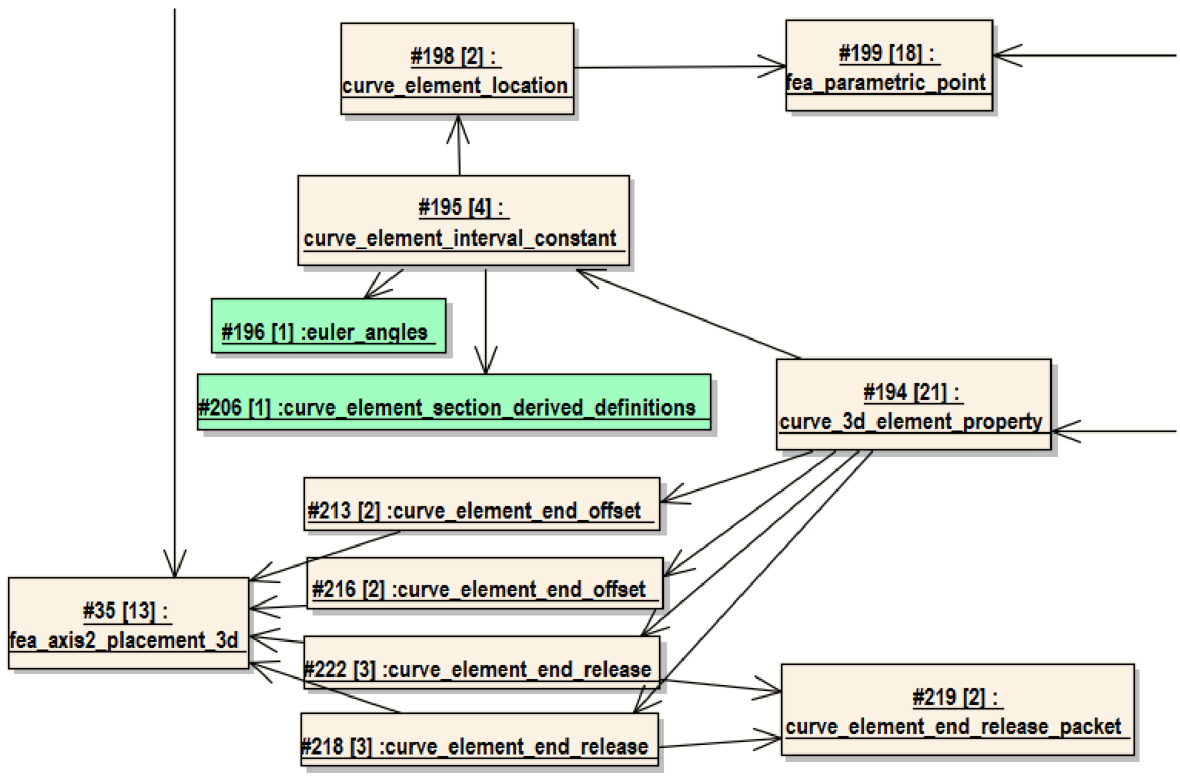


Figure 30 Details of Curve 3D Element Property

5.6.2 FE Model Input

The PROD property is defined and shared by all the elements in this pilot model. The input supplied for this model is shown in Figure 31.

1	2	3	4	5	6	7	8	9	10
PROD	PID	MID	A	J	C	NSM			

```
PROD      1         1         8.        0.
```

Figure 31 NASTRAN Card Image for Rod Property

The rod has a cross sectional area of 8. length units squared. In this problem the implied unit is inch. No assumption about cross sectional geometry is to be implied. The torsional constant 'J' has been set to 0 which will result in no torsional stiffness between the element grid points. The 'C' field is a coefficient used to compute torsional stress from the torsional moment and 'J'. Lastly, the 'NSM' field represents non-structural mass that for a 1-dimensional element translates into additional mass per unit length of the element.

5.6.3 STEP Entities

The AP209 ed2 representation of the input data shown in Figure 31 is listed in Figure 32.

```
#194= CURVE_3D_ELEMENT_PROPERTY('PROD.1','',(#195),(#213,#216),(#218,#222));

#195= CURVE_ELEMENT_INTERVAL_CONSTANT(#198,#196,#206);
#198= CURVE_ELEMENT_LOCATION(#199);
#199= FEA_PARAMETRIC_POINT('',(1.,0.,0.));
#196= EULER_ANGLES((0.,0.,0.));

#206= CURVE_ELEMENT_SECTION_DERIVED_DEFINITIONS(
        '',
        0.,
        8.,
        (CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),
        (0.,0.,0.),
        0.,
        CONTEXT_DEPENDENT_MEASURE(0.),
        (CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),
        (CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),
        (CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),
        CONTEXT_DEPENDENT_MEASURE(0.),
        CONTEXT_DEPENDENT_MEASURE(0.)
);

#213= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#216= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));

#218= CURVE_ELEMENT_END_RELEASE(#35,(#219));
#222= CURVE_ELEMENT_END_RELEASE(#35,(#219));
#219= CURVE_ELEMENT_END_RELEASE_PACKET(ENUMERATED_CURVE_ELEMENT_FREEDOM(.NONE.),0.);
```

Figure 32 STEP File Fragment for Curve 3D Element Property Structure (PROD)

The curve_3d_element_property entity is referenced by each element and in turn references all the particular instances that contribute to this property representation. Since the CROD only supports uniform properties throughout the element (beginning to end), a single constant interval is used. Multiple intervals could be used for stepped multi-section beams or multi-section tapered beams, but in this case there is only one constant section. Hence the use of a single curve_element_interval_constant reference in the curve_3d_element_property.

An interval is defined only by its endpoint so a single parametric point defines the interval as ending at the +1.0 parametric location. There is no twist specified so the euler_angle is 0.

The cross section property definition is specified on this interval. Again, only a single section definition is specified and many of the attributes are unspecified due to the simplicity of the element type. The listing above is formatted for readability and should match the EXPRESS listing shown in Figure 33.

```
ENTITY curve_3d_element_property;
        property_id                          : IDENTIFIER;
        description                          : TEXT;
        interval_definitions                 : LIST [1:?] OF curve_element_interval;
        end_offsets                          : ARRAY [1:2] OF curve_element_end_offset;
        end_releases                         : ARRAY [1:2] OF curve_element_end_release;
END_ENTITY;

ENTITY curve_element_section_derived_definitions;
    ENTITY curve_element_section_definition;
        description                          : TEXT;
        section_angle                        : PLANE_ANGLE_MEASURE;
    ENTITY curve_element_section_derived_definitions;
        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;

ENTITY curve_element_end_offset;
        coordinate_system                    : CURVE_ELEMENT_END_COORDINATE_SYSTEM;
        offset_vector                        : ARRAY [1:3] OF CONTEXT_DEPENDENT_MEASURE;
END_ENTITY;

ENTITY curve_element_end_release;
        coordinate_system                    : CURVE_ELEMENT_END_COORDINATE_SYSTEM;
        releases                             : LIST [1:?] OF curve_element_end_release_packet;
END_ENTITY;

ENTITY curve_element_end_release_packet;
        release_freedom                      : CURVE_ELEMENT_FREEDOM;
        release_stiffness                    : CONTEXT_DEPENDENT_MEASURE;
END_ENTITY;
```

Figure 33 EXPRESS Schema for Curve 3D Element Property and Related Entities

The offset and release entities do not apply to this element type but are required so the value 0. is used. The release packet uses the .NONE. enumeration for the release freedom to indicate that this characteristic does not apply to this element property. The interpretation of which end of the element that an instance of end offset or end release applies to, is determined by the order in the array of end offsets and array of end releases in the curve_3d_element_property.

5.7 Material Definition

The ROD element in NASTRAN only supports isotropic element usage so a single instance of the NASTRAN MAT1 element type was used in the pilot model. This material represents typical aluminum property values which implies a unit system of inch-pound force-second. However, the units in this STEP file instance do not reflect this implied system. When processing solver files that do not explicitly specify units, an external mechanism should be employed to augment the reading of the source model information.

5.7.1 STEP Diagram

The material data model in AP209 ed2 is complex. However, the isotropic material model used for the pilot models is the simplest instantiation possible for a 3 dimensional problem. The overall diagram of the material property model is shown in Figure 34.

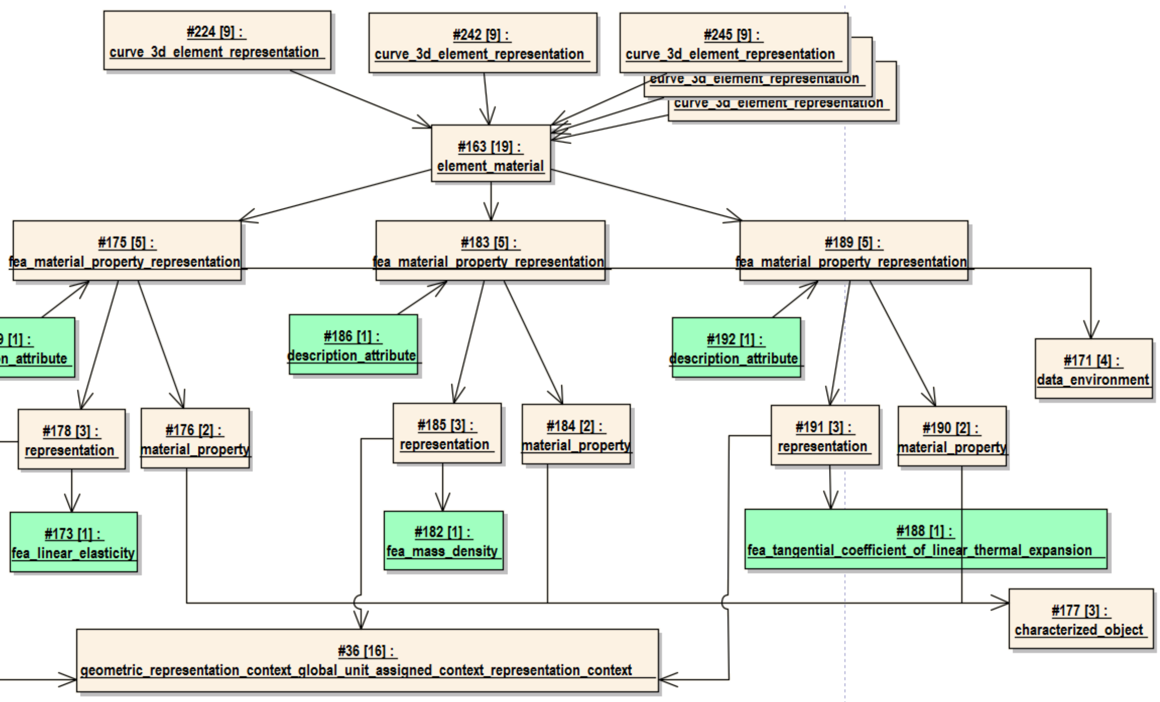


Figure 34 Overall Structure of Element Material Property

This figure shows the elements referring directly to a single element_material instance. This element material in turn references 3 fea_material_property_representations that form the set of properties aggregated into this element_material instance. Each of these generic entities also references a generic representation instance. This representation has an items list that is populated with a single instance of an fea_material_property_representation_item entity. This super-type is the parent of all the familiar material property entities used to package the material constants into the proper tensor type (green entities #173, #182, #188). The FEM input and EXPRESS schema fragments presented in the following sections will further explain the use of super-type sub-type and SELECT type for material property modeling.

Each of the fea_material_property_representation_item instances also refers to a single data_environment that defines the conditions under which the material constants were determined. Information such as temperature or moisture conditions can be specified under this entity. Additional data_environments can be defined if needed.

5.7.2 FE Model Input

The isotropic linear material is specified as shown in Figure 35.

1	2	3	4	5	6	7	8	9	10
MAT1	**MID**	**E**	**G**	**NU**	**RHO**	**A**	**TREF**	**GE**	
	ST	**SC**	**SS**	**MCSID**					

```
MAT1      1        1.+7            .33      2.54-4  1.3-5   70.
```

Figure 35 NASTRAN Card Image for Isotropic Material

The "E" field represents the modulus of elasticity, whereas, G represents the shear modulus. NU is Poisson's ratio, A is the coefficient of thermal expansion and TREF is the reference strain free temperature. The other solver specific fields are not used for this pilot model but are described in the NASTRAN documentation available on-line.

This input also illustrates a quirk of NASTRAN history. To maximize the number of significant digits possible in an 8-character field with a signed exponent, the use of the 'E' or 'e' preceding the exponent was eliminated. This can lead to erroneous parsing if not accounted for. Even "cut and paste" to familiar spread sheet tools can lead to unexpected results.

5.7.3 STEP Entities

The simple one-line MAT1 input card encodes a tremendous amount of information. It makes specification of the material in the solver deck simpler but can lead to ambiguous interpretation. The STEP data model expresses this in an un-ambiguous way as shown in Figure 36.

```
#163= ELEMENT_MATERIAL('MAT1.1','Fea Material',(#175,#183,#189));

#175= FEA_MATERIAL_PROPERTY_REPRESENTATION(#176,#178,#171);
#179= DESCRIPTION_ATTRIBUTE('Elasticity',#175);
#176= MATERIAL_PROPERTY('MAT1.1.0',$,#177);
#177= CHARACTERIZED_OBJECT('material_property',$);
#178= REPRESENTATION('representation_id_2',(#173),#36);
#173= FEA_LINEAR_ELASTICITY('',FEA_ISOTROPIC_SYMMETRIC_TENSOR4_3D((10000000.,0.33)));

#183= FEA_MATERIAL_PROPERTY_REPRESENTATION(#184,#185,#171);
#186= DESCRIPTION_ATTRIBUTE('MassDensity',#183);
#184= MATERIAL_PROPERTY('MAT1.1.1',$,#177);
#177= CHARACTERIZED_OBJECT('material_property',$);
#185= REPRESENTATION('representation_id_3',(#182),#36);
#182= FEA_MASS_DENSITY('',0.000254);

#189= FEA_MATERIAL_PROPERTY_REPRESENTATION(#190,#191,#171);
#192= DESCRIPTION_ATTRIBUTE('TangentCTE',#189);
#190= MATERIAL_PROPERTY('MAT1.1.2',$,#177);
#177= CHARACTERIZED_OBJECT('material_property',$);
#191= REPRESENTATION('representation_id_4',(#188),#36);
#188= FEA_TANGENTIAL_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION('',ISOTROPIC_SYMMETRIC_TENSOR2_3D(1.30E-5));
```

Figure 36 STEP File Fragment for Element Material

The element material entity references 3 material properties that are grouped together in Figure 36. Each group is defined by an instance of an fea_material_property_representations. The name and description attributes are useful for reading or displaying information about a single property, but are not part of the formal data identification. The existence of the fea_linear_elasticity, the fea_mass_density and the fea_tangential_coefficient_of_linear_thermal_expansion instances at the bottom of each property grouping as shown in Figure 37 is the proper way to identify what each fea_material_property_representation represents.

```
  ENTITY element_material;
          material_id                        : IDENTIFIER;
          description                        : TEXT;
          properties                         : SET [1:?] OF material_property_representation;
  END_ENTITY;

  ENTITY fea_material_property_representation;
      ENTITY property_definition_representation;
          definition                         : REPRESENTED_DEFINITION;
          used_representation                : representation;
        DERIVE
          description                        : TEXT;
          name                               : LABEL;
      ENTITY material_property_representation;
          dependent_environment              : data_environment;
      ENTITY fea_material_property_representation;
  END_ENTITY;

TYPE Represented_Definition = SELECT
  ( General_Property,
   Property_Definition,
   Property_Definition_Relationship,
   Shape_Aspect,
   Shape_Aspect_Relationship );
END_TYPE;

  ENTITY material_property;
      ENTITY property_definition;
          name                               : LABEL;
          description                        : OPTIONAL TEXT;
          definition                         : CHARACTERIZED_DEFINITION;
        DERIVE
          id                                 : IDENTIFIER;
      ENTITY material_property;
  END_ENTITY;
```

Figure 37 EXPRESS Schema for Element Material

The ‘properties’ attribute of the element_material is a set of material_property_representations. Since the fea_material_property_representation is a sub-type of this entity type, they are valid members of the set of properties for the element_material. The full list of possible entity types for these specific properties are found in the ‘fea_material_property_representation_item’ sub-types shown in Figure 38.

```
ENTITY Fea_Material_Property_Representation_Item
  SUPERTYPE OF (ONEOF (
          Fea_Linear_Elasticity,
          Fea_Mass_Density,
          Fea_Area_Density,
          Fea_Tangential_Coefficient_Of_Linear_Thermal_Expansion,
          Fea_Secant_Coefficient_Of_Linear_Thermal_Expansion,
          Fea_Moisture_Absorption,
          Fea_Shell_Membrane_Stiffness,
          Fea_Shell_Bending_Stiffness,
          Fea_Shell_Membrane_Bending_Coupling_Stiffness,
          Fea_Shell_Shear_Stiffness ) )
  SUBTYPE OF ( Representation_Item );
END ENTITY;
```

Figure 38 Valid Material Properties for Element Material

There is a pattern to how these entities at the bottom of the grouping are defined in the EXPRESS schema that facilitates understanding the structure. This pattern introduces a new concept in the EXPRESS schema, the 'SELECT' type. A 'SELECT' type allows a type to be any of a list of possible types or entities. Using the fea_linear_elasticity as an example is shown in Figure 39.

```
ENTITY fea_linear_elasticity;
      ENTITY representation_item;
          name                                    : LABEL;
      ENTITY fea_material_property_representation_item;
      ENTITY fea_linear_elasticity;
          fea_constants                           : SYMMETRIC_TENSOR4_3D;
END_ENTITY;

TYPE Symmetric_Tensor4_3d = SELECT
  ( Anisotropic_Symmetric_Tensor4_3d,
   Fea_Isotropic_Symmetric_Tensor4_3d,
   Fea_Iso_Orthotropic_Symmetric_Tensor4_3d,
   Fea_Transverse_Isotropic_Symmetric_Tensor4_3d,
   Fea_Column_Normalised_Orthotropic_Symmetric_Tensor4_3d,
   Fea_Column_Normalised_Monoclinic_Symmetric_Tensor4_3d );
END_TYPE;

TYPE Fea_Isotropic_Symmetric_Tensor4_3d = ARRAY [1 : 2] OF Context_Dependent_Measure;
END_TYPE;
```

Figure 39 Entity -> SELECT Type -> Type Pattern for Material Properties

The tensor type for this material property is a symmetric_tensor4_3d which dictates which mathematical operations are required to transform information in this tensor to a different geometric frame of reference. However, there are many simplifications of the full tensor expression that can be made based on the type of material being modeled. For this pilot model, an isotropic linear material reduces to 2 independent constants. Therefore, the fea_isotropic_symmetric_tensor4_3d type was selected. This type is a simple array of 2 context_dependent_measures (for Young's modulus (E) and Poisson's ratio (nu)).

The other properties are specified with similar patterns when tensor quantities are required. For example, the coefficient of linear expansion property is defined by the schema shown in Figure 40.

```
ENTITY fea_tangential_coefficient_of_linear_thermal_expansion;
      ENTITY representation_item;
          name                                    : LABEL;
      ENTITY fea_material_property_representation_item;
      ENTITY fea_tangential_coefficient_of_linear_thermal_expansion;
          fea_constants                           : SYMMETRIC_TENSOR2_3D;
END_ENTITY;

TYPE Symmetric_Tensor2_3d = SELECT
  ( Isotropic_Symmetric_Tensor2_3d,
   Orthotropic_Symmetric_Tensor2_3d,
   Anisotropic_Symmetric_Tensor2_3d );
END_TYPE;

TYPE Isotropic_Symmetric_Tensor2_3d = Context_Dependent_Measure;
END_TYPE;
```

Figure 40 EXPRESS Schema for Coefficient of Linear Thermal Expansion

The mass density property is different in that it is not a tensor quantity and is represented by a simple scalar data type which is a single context_dependent_measure (see Figure 41).

```
ENTITY fea_mass_density;
    ENTITY representation_item;
        name                                   : LABEL;
    ENTITY fea_material_property_representation_item;
    ENTITY fea_mass_density;
        fea_constant                           : SCALAR;
END_ENTITY;

TYPE Scalar = Context_Dependent_Measure;
END_TYPE;
```

Figure 41 EXPRESS Schema for Mass Density

Following the EXPRESS schema methodology for typing and sub-typing is not straight forward and probably beyond what this handbook is intended to convey. Readers should study the recommended practices and SMRL full EXPRESS listing to investigate further how these material properties are constructed. The most import aspect of this section is the way the tensor quantities are packaged into simple arrays in the AP209 ed2 data model. This topic is discussed in further detail in the AP209 ed2 recommended practices document and Part 104 Section 7.

Lastly, the data_environment for the element_material property items is diagrammed in Figure 42.

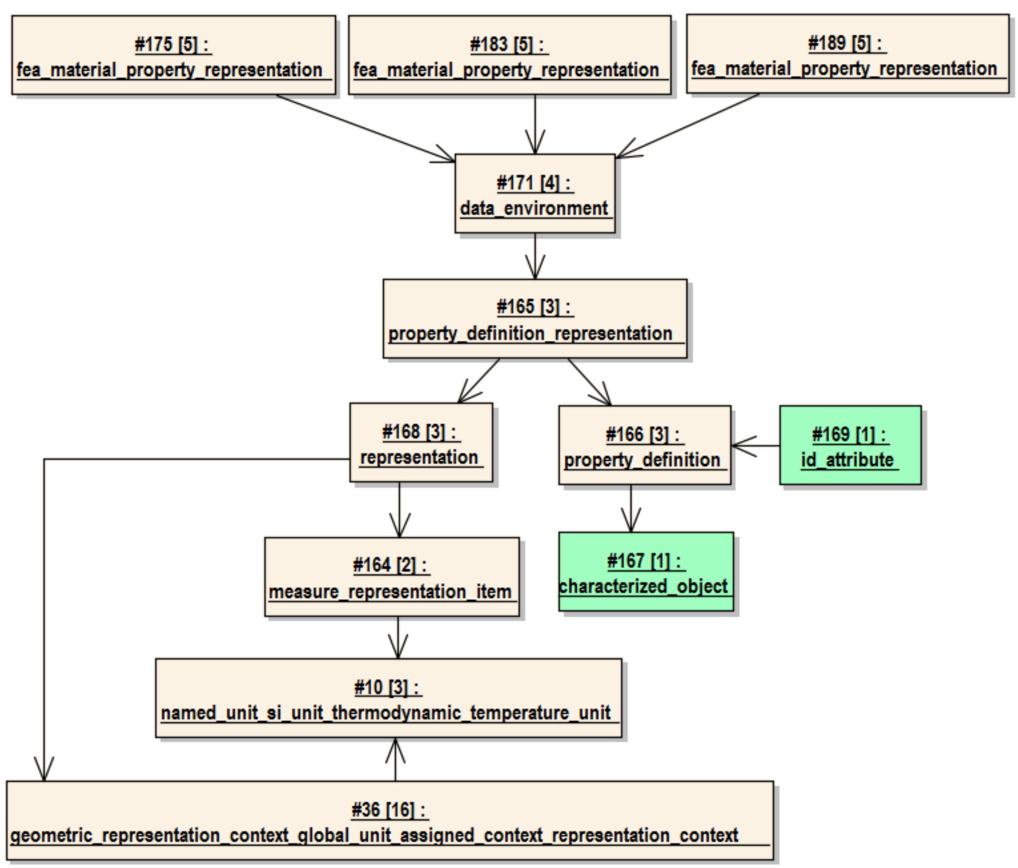


Figure 42 STEP Data Environment Instance Diagram

Figure 42 starts with the three fea_material_property_representations that were discussed above. It uses a slightly different property specification methodology that does not involve the SELECT typing and tensor packaging discussed in Section 5.7.3. Instances of property_definition_representation(s) are listed in the items list for the data_environment entity. However, for this pilot model, temperature is the only environmental property specified so there is only one property_definition_representation listed. The property_definition_representation uses a representation instance to specify a measure_representation_item in the items list. The measure_representation_item is where the property value is captured as a measure and also provides a direct link to a unit as appropriate for the property.

The listing shown in Figure 43 illustrates the data_environment used for this pilot model. Both representational and definitional aspects of the properties are supported. *Note (as of this writing, the temperature measure is correct but the specified unit .DEGREE_CELSIUS. is incorrect. Future releases of this handbook will correct this error)*

```
#171= DATA_ENVIRONMENT('DATA_ENV','Property_conditions',(#165));

#165= PROPERTY_DEFINITION_REPRESENTATION(#166,#168);
#168= REPRESENTATION('representation_id_1',(#164),#36);
#164= MEASURE_REPRESENTATION_ITEM('representation_item_name_1',THERMODYNAMIC_TEMPERATURE_MEASURE(70.),#10);

#166= PROPERTY_DEFINITION('Material Property Definition',$,#167);
#167= CHARACTERIZED_OBJECT('property',$);
#169= ID_ATTRIBUTE('MAT1.1',#166);

#10= /* SI_UNIT+THERMODYNAMIC_TEMPERATURE_UNIT */
(NAMED_UNIT(*)SI_UNIT($,.DEGREE_CELSIUS.)THERMODYNAMIC_TEMPERATURE_UNIT());
```

Figure 43 STEP File Fragment of Data Environment

The property_definition (from the data_environment) and the material_property (from the fea_material_property_representations) provide a reference to additional information about the property if needed. Currently, the characterized_object is a placeholder for the additional information. This link could be used to establish traceability to very detailed information provided by companion standards such as AP235 (10303-235) or to individual material products. An example of how Figure 34 would look if a reference to an isotropic material product instance was used is shown in Figure 44. Material products are required if laminate tables for composite materials are used.

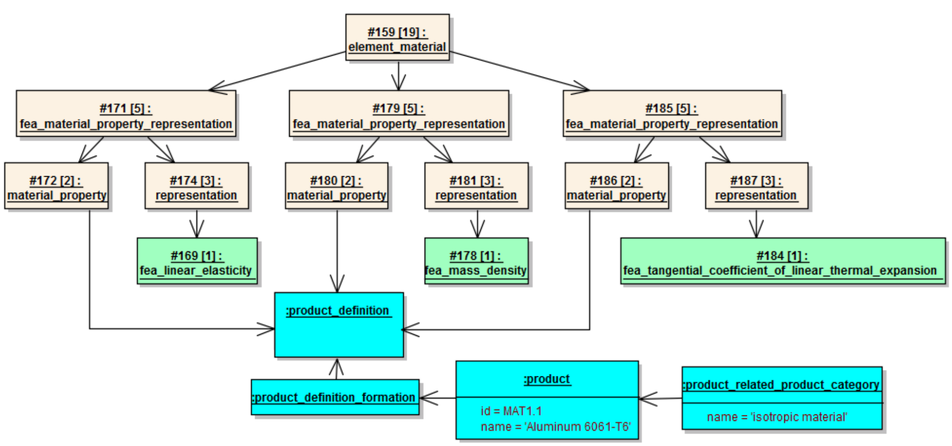


Figure 44 Use of Material Product and Product_definition in place of Characterized_object

The EXPRESS schema shown in Figure 45 for the data_environment, finalizes the material property discussion. The last EXPRESS entity definition for measure_representation_item introduces the EXPRESS construct of multiple inheritance. This construct defines an entity that is a union of both the listed super-types. This creates an instance that can fill the role of both super-types in the STEP data model.

```
ENTITY data_environment;
        name                        : LABEL;
        description                 : TEXT;
        elements                    : SET [1:?] OF property_definition_representation;
END_ENTITY;

ENTITY property_definition_representation;
        definition                          : REPRESENTED_DEFINITION;
        used_representation                 : representation;
      DERIVE
        description                         : TEXT;
        name                                : LABEL;
END_ENTITY;

TYPE REPRESENTED_DEFINITION = SELECT
     (general_property,
      property_definition,
      property_definition_relationship,
      shape_aspect,
      shape_aspect_relationship);
END_TYPE;

ENTITY property_definition;
        name                                : LABEL;
        description                         : OPTIONAL TEXT;
        definition                          : CHARACTERIZED_DEFINITION;
      DERIVE
        id                                  : IDENTIFIER;
END_ENTITY;

ENTITY Measure_Representation_Item
   SUBTYPE OF ( Representation_Item, Measure_With_Unit );
END_ENTITY;
```

Figure 45 EXPRESS Schema for Data Environment

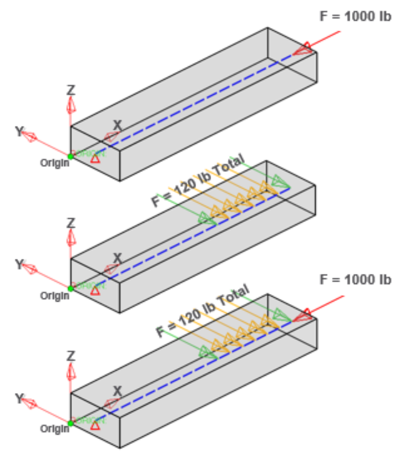
A measure_representation_item is both a representation_item and measure_with_unit where a numerical measure value and its unit is defined. The definition of measure_with_unit is a super-type that list the 33 physical quantity types found in the SI Coherent system of units (see Part 41 and ISO 31). In this case, a thermodynamic_temperature_measure_with_unit has a unit_component of thermodynamic_temperature_unit and value_component that is a thermodynamic_measure type with a real value.

6 Pilot Model ATS2 – 1D Bar Element

The ATS2 pilot model composition is illustrated in the Figure 46. It is a variation of the ATS1 pilot model in that a different solver element type is used to represent more complex behaviors by using additional degrees of freedom.

- Bar with axial and bending stiffness (no torsional stiffness)
- FE model composition

Elements:	16	CBAR
Nodes:	17	GRID
Loads:	8	FORCE
	2	LOAD
Boundary:	1	SPC1
	1	SPCADD
Property:	1	PBAR
Material:	1	MAT1 (aluminum)
System:	1	CORD2R (at origin)



- Subcase and output requests

Subcases:	3	SUBCASE
Boundary:	3	SPC
Loads:	3	LOAD
Output:	12	GPFORCE (global) DISPLACEMENT SPCFORCES STRESS

See listing for output parameters

Figure 46 Pilot Model Content Summary for ATS2

Additional details of the model are:

- Introduction of boundary condition combinations
- A lateral bending load case as well as a scaled linear combination load case
- Exercise the double field NASTRAN card format in the input file
- All grid points are defined in a single local Cartesian coordinate system id=1

6.1 Product Definition Shape

Since this pilot model uses the same local coordinate system as ATS1, the diagram shown in Figure 12 for the ATS1 pilot model is valid with the exception that the instance IDs may have changed. The discussion of section 5.1 applies in its entirety to this pilot model.

6.2 FEA Model

The discussion in Section 5.2 applies to the ATS2 model. As shown in Figure 47, the fea_model_3d instance is identical to the ATS1 model.

```
#77= FEA_MODEL_3D('Identification',(#35),#36,
'NASTRAN BDF Converter v0.0.0',('NASTRAN'),'AnalysisModelType');
```

Figure 47 STEP File Fragment for FEA Model 3D

6.3 Coordinate System

The coordinate system discussion in Section 5.3 applies.

6.3.1 STEP Diagram

The instance diagram in Section 5.3.1 is identical for this model.

6.3.2 FE Model Input

The FEA model input for coordinate systems in model ATS2 is identical to Section 5.3.1.

6.3.3 STEP Entities

The STEP instances for coordinate systems are identical to Section 5.3.3 and are listed in Figure 48.

```
#35= FEA_AXIS2_PLACEMENT_3D('0',#33,#31,#29,.CARTESIAN.,'FEA_BASIC_COORD_SYSTEM.0');

#36= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT('FEA_BASIC_COORD_SYSTEM.0','3d'));

#29= DIRECTION('RefX',(1.,0.,0.));
#31= DIRECTION('AxisZ',(0.,0.,1.));
#33= CARTESIAN_POINT('',(0.,0.,0.));

#45= FEA_AXIS2_PLACEMENT_3D('1',#43,#41,#39,.CARTESIAN.,'CORD2R.1');

#46= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT('CORD2R.1','3d'));

#39= DIRECTION('RefX',(1.,0.,0.));
#41= DIRECTION('AxisZ',(0.,0.,1.));
#43= CARTESIAN_POINT('',(0.,0.,0.));

#92= POINT_REPRESENTATION('point_representation_name_1',(#93,#45),#46);
#93= NODE_SET('node_set_name_1',(#90,#99,#103,#107,#111,#115,#119,#123,
#127,#131,#135,#139,#143,#147,#151,#155,#159));

#161= ITEM_DEFINED_TRANSFORMATION('','',#35,#45);;
#162= REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION('','',#77,#92,#161);
```

Figure 48 STEP File Fragment for Basic and Local Coordinate Systems

6.4 Node Definition

The grid point discussion in Section 5.4 applies.

6.4.1 STEP Diagram

The diagram shown in Section 5.4.1 applies; however, some instance IDs may be different.

6.4.2 FE Model Input

The input from the FEA model is identical to Section 5.4.2.

6.4.3 STEP Entities

The node instances are identical to Section 5.4.3; however, the node_group has a different ID. Figure 49 shows the STEP entities for the first 3 nodes and the node_group instance.

```
#90= NODE('1',(#88),#46,#77);
#88= CARTESIAN_POINT('1',(0.,-2.,1.));

#99= NODE('2',(#97),#46,#77);
#97= CARTESIAN_POINT('2',(1.,-2.,1.));

#103= NODE('3',(#101),#46,#77);
#101= CARTESIAN_POINT('3',(2.,-2.,1.));


#625= NODE_GROUP('','ALL',#77,(#90,#99,#103,#107,#111,#115,#119,#123,
#127,#131,#135,#139,#143,#147,#151,#155,#159));
#626= ID_ATTRIBUTE('ALL',#625);
```

Figure 49 STEP File Fragment for Nodes and Node_Group

6.5 Element Definition

A linear bar element is defined between each grid point to supply stiffness to the grid points. Bar elements are capable of supplying stiffness to the full 6 degrees of freedom at each grid point. Again, this discussion is related to the definition of the FEA solver element and not the AP209 ed2 entity of the same name. The same curve_3d_element_representation entity is used as for the pilot model ATS1 but the property definitions are populated differently as is the element descriptor to reflect the additional capabilities provided by this solver element type.

6.5.1 STEP Diagram

The structure for a single element is identical to Section 5.5.1, but the related instances that can be shared from one element to another, such as element property, are different. Figure 50 illustrates that difference.

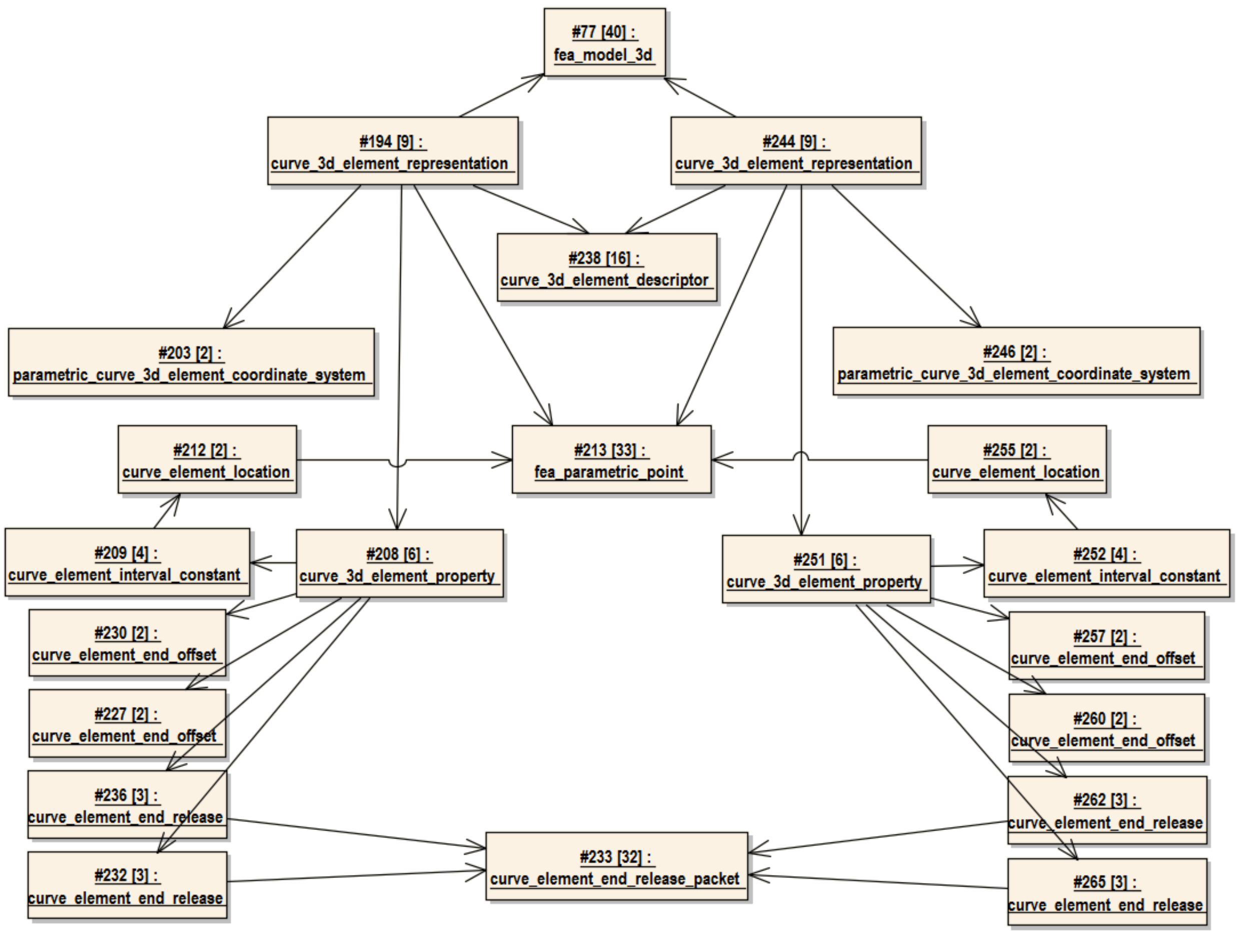


Figure 50 Structure of Curve_3d_element_representation for ATS2

Many of the common instances have been hidden in this diagram. The main difference of note is that because NASTRAN CBAR elements support individual offsets and end releases defined on each element, there could exist a unique curve_3d_element_property for each element. Those properties must then reference unique curve_element_interval_constants and individual end releases and offsets. A single fea_parametric_point can be used since there are no multi section bars in this model. A single instance of the curve_element_end_release_packet is also used. This is possible because all end releases on all elements are blank and the single instance of curve_element_end_release_packet is populated with a freedom enumeration equal to '.NONE.'. This is discussed further in Section 6.

Additional optimization of the STEP output could be performed on the curve_3d_element_property instances to reuse more instances for common element property IDs. To take advantage of this optimization, key or hashing schemes should be employed to determine equality of the combined interval definition, property id, end release and end offset information as found in the parsed solver input. Optimization of output is a complex topic that is only introduced in this document and is an active area of discussion and development in the STEP community.

6.5.2 FE Model Input

There are 16 NASTRAN elements defined using the CBAR input card shown in Figure 51. The bar elements used in this pilot model supply stiffness for all 6 degrees of freedom at each grid point. CBAR elements can reference property cards that either define the beam section constants directly (PBAR) or define a geometric cross section shape (PBARL). The PBAR property is used for the initial pilot models.

1	2	3	4	5	6	7	8	9	10
CBAR	EID	PID	GA	GB	X1	X2	X3	OFFT	
	PA	PB	W1A	W2A	W3A	W1B	W2B	W3B	

```
CBAR*    1                1                1                2
*        0.               7.54979-8        1.
CBAR*    2                1                2                3
*        0.               7.54979-8        1.
```

Figure 51 NASTRAN Input Specification for 1D CBAR Element

Another variation introduced with this element type is the use of double-field input where each logical field is physically 16 characters long. Therefore, two lines of input are required to express each line of input in the card diagrams shown.

Additionally, an orientation vector (X1, X2, X3) is introduced that necessitates individual parametric_curve_3d_element_coordinate_systems. This orientation vector clocks the local element coordinate system to locate the reference directions for complex section properties such as those shown in Figure 52. The orientation vector data also contains a very small, near zero, real value. This value was included to test small value handling during the translation process.

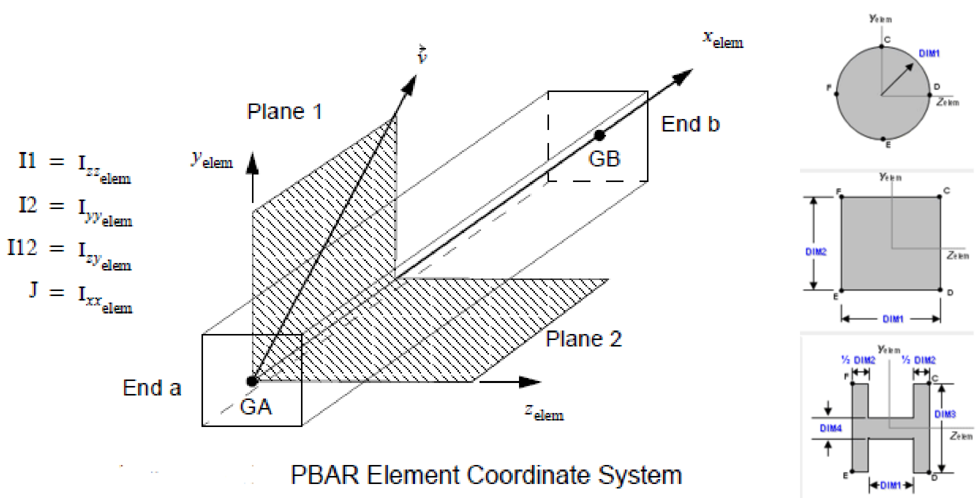


Figure 52 Element Coordinate System for NASTRAN CBAR Property Reference

6.5.3 STEP Entities

The 1-dimensional bar element maps to the same curve_3d_element_representation entity as the ATS1 rod element model. The behavioral aspects are specified differently than in Section 5.5.3 through use of a curve_3d_element_descriptor with additional enumerated purposes defined. The additional behaviors account for the bending and shear capabilities of the NASTRAN CBAR element. Again, the descriptor text has been formatted for readability and generally will not appear as shown by the STEP files in Figure 53.

```
#194= CURVE_3D_ELEMENT_REPRESENTATION('1',(#203,#213),#195,(#90,#99),#77,#238,#208,#163);
#244= CURVE_3D_ELEMENT_REPRESENTATION('2',(#246,#213),#195,(#99,#103),#77,#238,#251,#163);

"#238= CURVE_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_CURVE.CBAR*',(
(ENUMERATED_CURVE_ELEMENT_PURPOSE(.AXIAL.)),
(ENUMERATED_CURVE_ELEMENT_PURPOSE(.Y_Y_BENDING.),
 ENUMERATED_CURVE_ELEMENT_PURPOSE(.Z_Z_BENDING.)),
(ENUMERATED_CURVE_ELEMENT_PURPOSE(.TORSION.)),
(ENUMERATED_CURVE_ELEMENT_PURPOSE(.X_Y_SHEAR.),
ENUMERATED_CURVE_ELEMENT_PURPOSE(.X_Z_SHEAR.))
));"
```

Figure 53 STEP Entities for CBAR Element and Element Descriptor

The local element coordinate system instances referenced by the two element representations above are shown in Figure 54. Again, there are opportunities for optimization of the STEP output using the orientation values to determine equality of the direction instances.

```
#203= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#204);
#204= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#205);
#205= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));

#246= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#247);
#247= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#248);
#248= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));

#213= FEA_PARAMETRIC_POINT('',(1.,0.,0.));

#195= /* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CONTEXT()
REPRESENTATION_CONTEXT('REP_CONTEXT_FEA','3D'));
```

Figure 54 CBAR Parametric Element Coordinate System and Context

6.6 Property Definition

The property definition for the ATS2 pilot model follows a similar structure as for the ATS1 model with the exception of the possibility of unique property instances to account for information that may be specified on individual element. This topic was touched upon in Section 6.5.1 and is not repeated here. The indexing and retrieval schemes that are employed to take advantage of instance reuse are implementation dependent and beyond the intent of this pilot model discussion. However, every effort should be taken to capture the modeling intent of the source FEA model construction when possible to assist with solver input reconstruction.

In situations that require multiple property entities to be created in the AP209 ed2 data model where only a single property was read from the source data, the property metadata such as name and description fields should be used to indicate the source association of the multiple properties. Typically, these types of weak relationship expressions are not to be relied upon for computer processing but are of great value to the end consumer of the FEA information and should be retained.

6.6.1 STEP Diagram

The diagram for a single property is identical to that shown in Section 5.6.1. However, there are multiple instances of curve_3d_element_property and many more terms in the section definition are non-zero to reflect the additional bending properties of the CBAR element.

6.6.2 FE Model Input

The NASTRAN input model shown in Figure 55 has only one PBAR defined which means that all the elements share the same section properties.

1	2	3	4	5	6	7	8	9	10
PBAR	PID	MID	A	I1	I2	J	NSM		
	C1	C2	D1	D2	E1	E2	F1	F2	
	K1	K2	I12						

```
PBAR     1       1       8.       2.667   10.667  0.      0.
         1.      2.      1.      -2.      0.      0.      0.      0.
                         0.
```

Figure 55 NASTRAN Input Specification for 1D PBAR

Similar to the ATS1 pilot model, the 'J' torsional term is specified as zero. The C and D non-zero fields are stress recovery locations in the plane of the cross section and are not used in this model.

6.6.3 STEP Entities

The STEP AP209 ed2 property listing for the two properties shown in the diagram are shown in Figure 56. The order displayed is different in that the common instances are listed at the end.

```
#208= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.1','',(#209),(#227,#230),(#232,#236));

#209= CURVE_ELEMENT_INTERVAL_CONSTANT(#212,#210,#220);
#212= CURVE_ELEMENT_LOCATION(#213);
#210= EULER_ANGLES((0.,0.,0.));
#227= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#230= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#232= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#236= CURVE_ELEMENT_END_RELEASE(#35,(#233));

#251= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.2','',(#252),(#257,#260),(#262,#265));

#252= CURVE_ELEMENT_INTERVAL_CONSTANT(#255,#253,#220);
#255= CURVE_ELEMENT_LOCATION(#213);
#253= EULER_ANGLES((0.,0.,0.));
#257= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#260= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#262= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#265= CURVE_ELEMENT_END_RELEASE(#35,(#233));

#233= CURVE_ELEMENT_END_RELEASE_PACKET(ENUMERATED_CURVE_ELEMENT_FREEDOM(.NONE.),0.);
#213= FEA_PARAMETRIC_POINT('',(1.,0.,0.));

#220= CURVE_ELEMENT_SECTION_DERIVED_DEFINITIONS('PBAR.1_0',
0.,
8.,
(CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),
(2.667,10.667,0.),
0.,
CONTEXT_DEPENDENT_MEASURE(0.),
(CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),
(CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),
(CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),
CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.));
```

Figure 56 STEP File Fragment for Curve 3D Element Property Structure (PBAR)

Again the topics of instance reuse and output optimization are mentioned due to the obvious repetition of identical instances in this property listing.

6.7 Material Definition

The material definition and discussion is identical to that presented in Section 5.7.

6.7.1 STEP Diagram

Identical to Section 5.7.1.

6.7.2 FE Model Input

Identical to Section 5.7.2

6.7.3 STEP Entities

The instances for the material definition shown in Figure 57 are identical to Section 5.7.3.

```
#163= ELEMENT_MATERIAL('MAT1.1','Fea Material',(#175,#183,#189));

#175= FEA_MATERIAL_PROPERTY_REPRESENTATION(#176,#178,#171);
#179= DESCRIPTION_ATTRIBUTE('Elasticity',#175);
#176= MATERIAL_PROPERTY('MAT1.1.0',$,#177);
#177= CHARACTERIZED_OBJECT('material_property',$);
#178= REPRESENTATION('representation_id_2',(#173),#36);
#173= FEA_LINEAR_ELASTICITY('',FEA_ISOTROPIC_SYMMETRIC_TENSOR4_3D((10000000.,0.33)));

#183= FEA_MATERIAL_PROPERTY_REPRESENTATION(#184,#185,#171);
#186= DESCRIPTION_ATTRIBUTE('MassDensity',#183);
#184= MATERIAL_PROPERTY('MAT1.1.1',$,#177);
#177= CHARACTERIZED_OBJECT('material_property',$);
#185= REPRESENTATION('representation_id_3',(#182),#36);
#182= FEA_MASS_DENSITY('',0.000254);

#189= FEA_MATERIAL_PROPERTY_REPRESENTATION(#190,#191,#171);
#192= DESCRIPTION_ATTRIBUTE('TangentCTE',#189);
#190= MATERIAL_PROPERTY('MAT1.1.2',$,#177);
#177= CHARACTERIZED_OBJECT('material_property',$);
#191= REPRESENTATION('representation_id_4',(#188),#36);
#188= FEA_TANGENTIAL_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION('',ISOTROPIC_SYMMETRIC_TENSOR2_3D(1.30E-5));
```

Figure 57 STEP Fragment for Isotropic Material Properties

7 Pilot Model ATS3 – 2D Shell Element

The ATS3pilot model composition is illustrated in Figure 58.

- Shell model with wider variety of boundary condition and load combinations
- FE model composition

Elements:	40	CQUAD4
	48	CTRAI3
Nodes:	85	GRID
Loads:	12	FORCE
	8	PLOAD2 (normal pressure)
	3	LOAD
Boundary:	104	SPC1
	2	SPCADD
Property:	1	PSHELL
Material:	1	MAT1 (aluminum)

- Subcase and output requests

Subcases:	4	SUBCASE
Boundary:	4	SPC
Loads:	4	LOAD
Output:	16	GPFORCE (global)
		DISPLACEMENT
		SPCFORCES
		STRESS

*See listing for output parameters

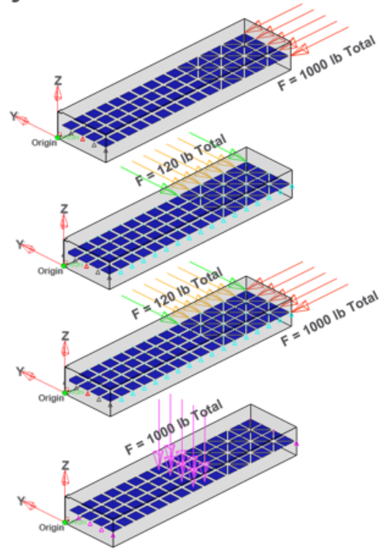


Figure 58 Pilot Model Content Summary ATS3

Additional details of the model are:

- All nodes defined in basic coordinate system, no local systems defined
- Shell element normal pressure load case (-z) direction
- Uses inclusive continuous node range definition in boundary condition specification
- Rich combinations of boundary conditions and load cases to model beam behaviors

7.1 Product Definition Shape

The structure of the product definition entity graph is shown in Figure 59 for the 2D shell element ATS3 pilot model. It is very similar to the previous pilot models except that there are no local coordinate systems.

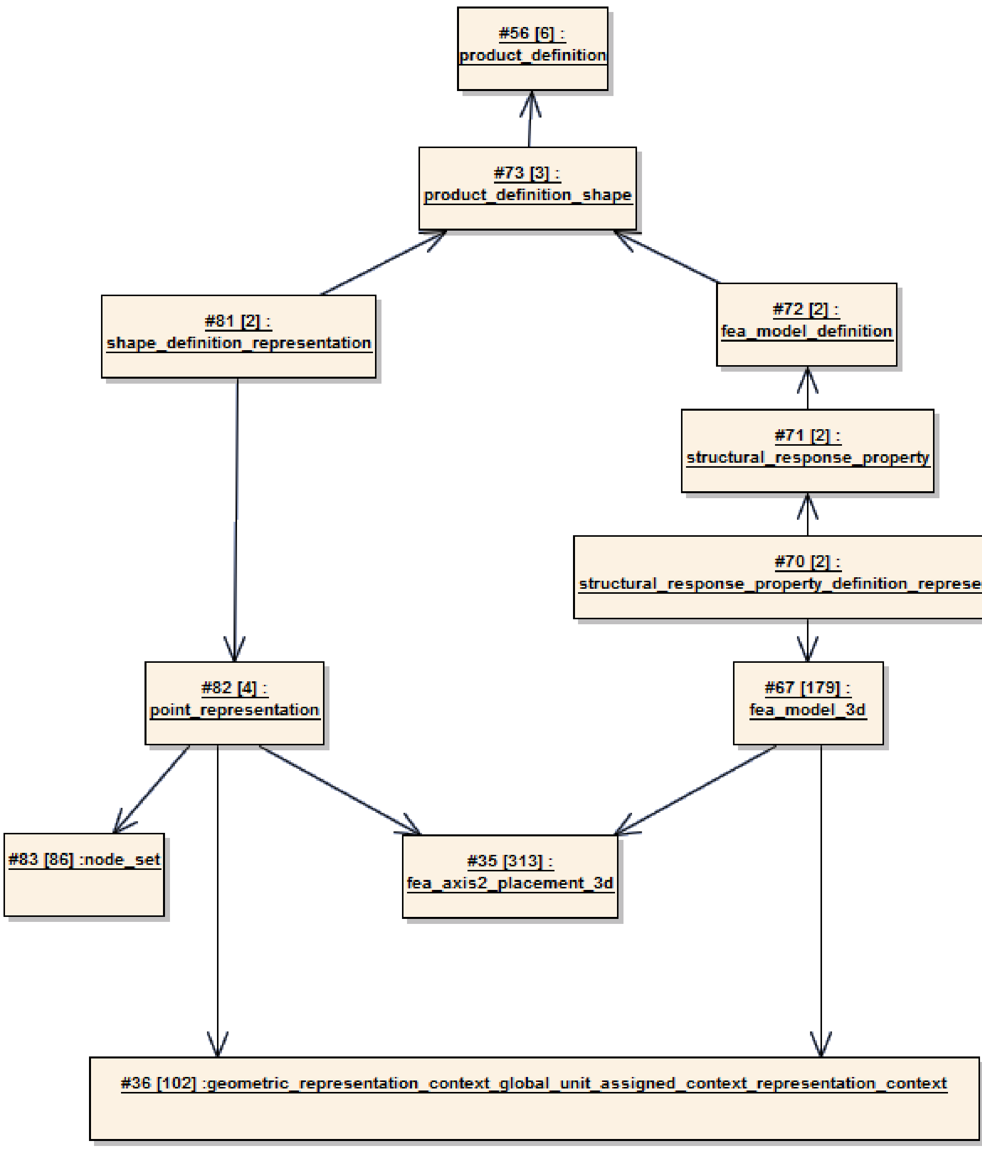


Figure 59 Basic coordinate system used for finite element model geometric entities

7.2 FEA Model

The fea_model_3d entity is unchanged from the prior examples. However, as noted above, a single fea_axis2_placement3d is shared between the fea_model_3D and the point representation. This indicates that all geometric entities related to the FEM are founded with respect to the implied basic NASTRAN coordinate system. No transformation operators are required.

7.3 Coordinate System

No local coordinate systems are defined for the ATS3 model. Only the basic coordinate system is used as shown in Figure 59. According to the recommended practice document, the point_representation and associated node_set are optional in this case since all nodes are founded with respect to the fea_model_3d and associated fea_axis2_placement_3d. However, for consistency, the point_representation and associated node_set are instantiated and associated as shown above. In this instance, the node_set will have a list of references to all the nodes.

7.3.1 STEP Diagram

No additional diagram needed for coordinate systems.

7.3.2 FE Model Input

The NASTRAN basic coordinate system is an implied system and no user input is required.

7.3.3 STEP Entities

The STEP entities used to define the basic coordinate system are shown in Figure 60 for reference. Note the use of the complex context entity. In this type of EXPRESS construct, multiple entities can be aggregated into a single instance to indicate an inseparable relationship and to provide a single reference to be used throughout the STEP model.

```
#35= FEA_AXIS2_PLACEMENT_3D('0',#33,#31,#29,.CARTESIAN.,'FEA_BASIC_COORD_SYSTEM.0');

#36= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
    (GEOMETRIC_REPRESENTATION_CONTEXT(3)
     GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,#12,#17,#7,#8,#11,#26,#20,#10,#9))
     REPRESENTATION_CONTEXT('FEA_BASIC_COORD_SYSTEM.0','3d'));

#29= DIRECTION('RefX',(1.,0.,0.));
#31= DIRECTION('AxisZ',(0.,0.,1.));
#33= CARTESIAN_POINT('',(0.,0.,0.));

#82= POINT_REPRESENTATION('point_representation_name_1',(#83,#35),#36);
```

Figure 60 STEP File Fragment for Basic Coordinate System

7.4 Node Definition

The nodes defined for the ATS3 model represent the same 16 inch prismatic shape as the prior models; however, the use of shell elements requires a 2-dimensional geometric layout of nodes. There are 85 geometric nodes defined in the model (17 rows of 5 nodes). All the nodes are located in a plane located at the mid-surface of the shape (Z=1.0).

7.4.1 STEP Diagram

The node instance diagram provided by Figure 61 can be compared with the one presented in Figure 19. Note the differences in these diagrams that reflect the different geometric founding. The nodes are related to the FEA model which in turn is related to the single instance of fea_axis2_placement_3d that represents the basic coordinate system. The node_set and point_representation provide a redundant path from all the nodes to the single instance of fea_axis2_placement_3d for geometric founding. This redundancy is why the point_representation and node_set are optional when only one coordinate system is used.

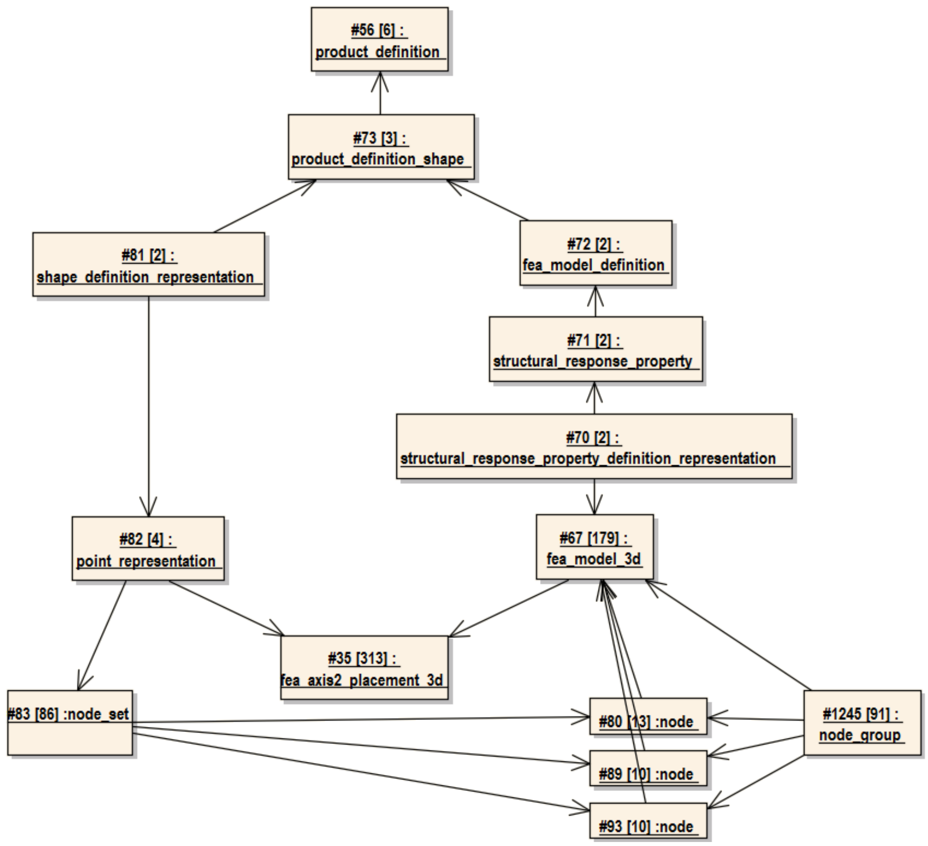


Figure 61 STEP Node Entity Relationships to other FEA Instances

7.4.2 FE Model Input

The nodes shown in Figure 61 are created as a result of the NASTRAN input shown in Figure 62. Note that both the 'CP' and 'CD' attributes are blank which implies use of the basic NASTRAN coordinate system.

1	2	3	4	5	6	7	8	9	10
GRID	ID	CP	X1	X2	X3	CD	PS	SEID	

```
GRID*    1                                 -6.24022-8      -4.
*        1.
GRID     2                1.      -4.      1.
GRID     3                2.      -4.      1.
```

Figure 62 NASTRAN Card Image for Grid Points

This input also illustrates the use of mixed format input in the NASTRAN source file. The first grid uses the 16-character wide data fields which results in a second line of input to define the grid point. The use of double field input is necessary to minimize numeric errors that arise when larger values differ by a small value. This scenario is common in vehicle modeling when coordinates are used for large vehicles with small detailed model features. In these cases, the use of the NASTRAN short field (8-character) format may cause numeric errors due to collapsed element definitions.

7.4.3 STEP Entities

The translated NASTRAN nodes are listed in Figure 63. The three references for each node instance represent the associated cartesian_point (#78 for node number 1), the 3-dimensional geometric_representation_context (#36), and the fea_model_3d (#67).

```
#80= NODE('1',(#78),#36,#67);
#78= CARTESIAN_POINT('1',(-6.240220000000000E-8,-4.,1.));

#89= NODE('2',(#87),#36,#67);
#87= CARTESIAN_POINT('2',(1.,-4.,1.));

#93= NODE('3',(#91),#36,#67);
#91= CARTESIAN_POINT('3',(2.,-4.,1.));

#83= NODE_SET('node_set_name_1',(#80,#89,#93,#97,#101,#105,#109,#113,
#117,#121,#125,#129,#133,#137,#141,#145,#149,#153,#157,#161,#165,#169,
#173,#177,#181,#185,#189,#193,#197,#201,#205,#209,#213,#217,#221,#225,
#229,#233,#237,#241,#245,#249,#253,#257,#261,#265,#269,#273,#277,#281,
#285,#289,#293,#297,#301,#305,#309,#313,#317,#321,#325,#329,#333,#337,
#341,#345,#349,#353,#357,#361,#365,#369,#373,#377,#381,#385,#389,#393,
#397,#401,#405,#409,#413,#417,#421));
```

Figure 63 STEP File Fragment for ATS3 Nodes

7.5 Element Definition

The element definition for ATS3 introduces 4-sided quadrilateral and 3-sided triangular shell elements. These elements can support 6 degrees of freedom at each node used in the definition of the element connectivity. This model uses a linear formulation for these shell elements which requires that only corner nodes be defined for each element. Higher-order element types are possible when additional nodes are used on element edges or faces. These types of elements support quadratic and cubic interpolation functions in the element formulation but are not used in this pilot study. Both the element shape and order are defined in the associated element descriptor instance.

7.5.1 STEP Diagram

The element diagram for the first two quadrilateral elements is shown in Figure 64. The relationship to the nodes for these two elements indicates that they are adjacent and share the edge defined by nodes 2 and 13 (STEP instance IDs #89 and #133). In the solver model, these elements use a common physical property and material property as indicated by the relationships to the surface_element_property and element_material instances.

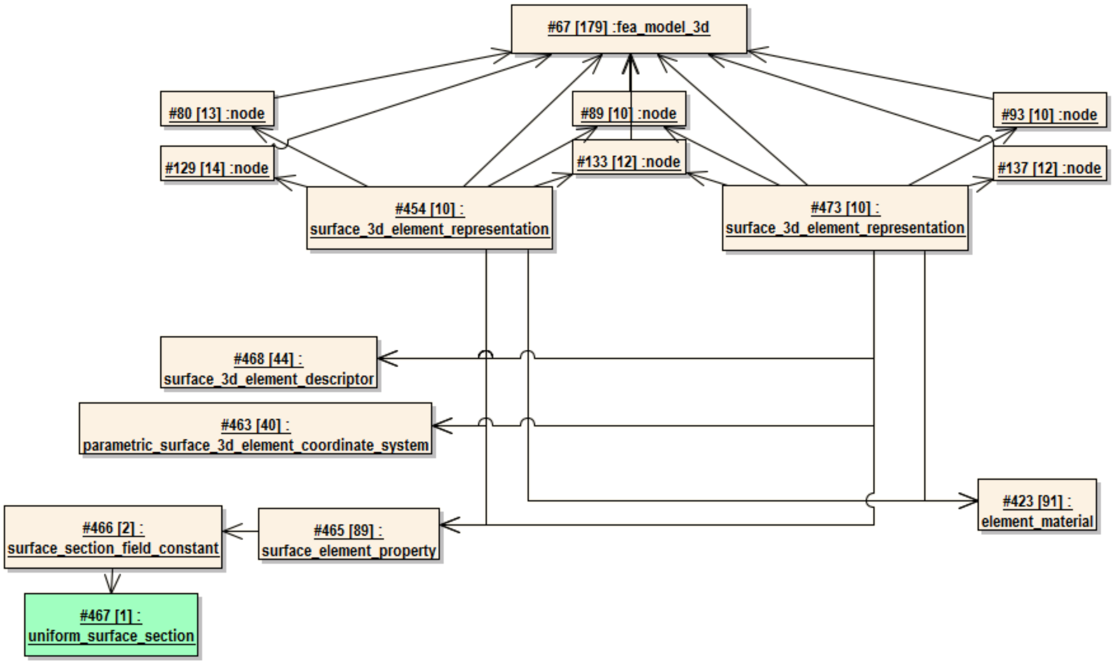


Figure 64 STEP Surface 2D Element Definition and Related Data Model Structures

A detail that may not be immediately apparent is the different number of associations indicated by the node label in brackets [] shown on this diagram (eg. node #89 has [10] associations and node #129 has [14]). This indicates that node #129 is referenced by additional entities such as loads or boundary conditions. It could also be due to differences in mesh topology such as an external corner node or interior node shared by many elements. There are many possibilities to explain different association counts. Figure 65 illustrates this point and shows the additional references on node #129 are in fact boundary conditions defined by single_point_constraint_elements.

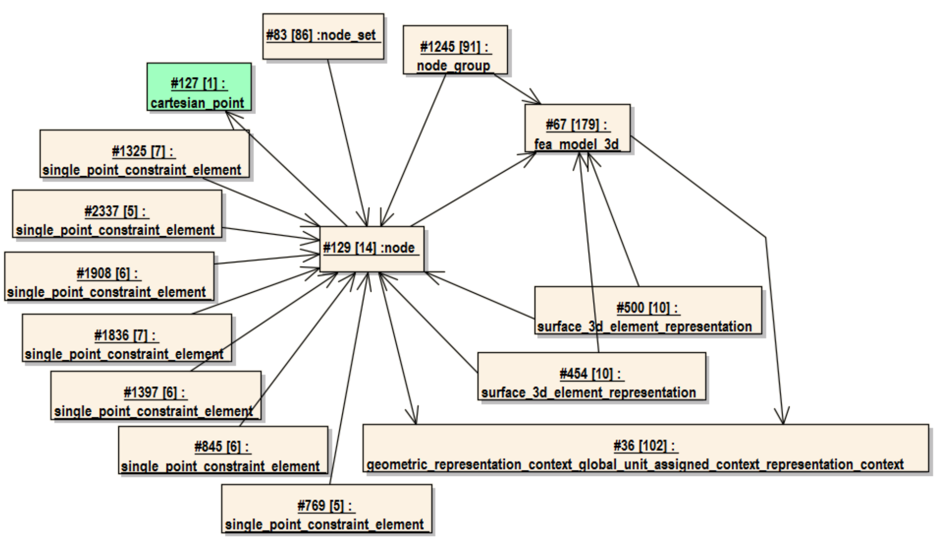


Figure 65 STEP Instances Associated to NODE #129

Similar figures could be constructed for the triangular element definitions with the primary difference being that only 3 nodes will be in the item_list for the surface_3d_element_representation.

7.5.2 FE Model Input

The NASTRAN input for shell elements is shown in Figure 66. Both quadrilateral and triangular element input and STEP listings are presented.

1	2	3	4	5	6	7	8	9	10
CQUAD4	EID	PID	G1	G2	G3	G4	THETA or MCID	ZOFFS	
		TFLAG	T1	T2	T3	T4			

1	2	3	4	5	6	7	8	9	10
CTRIA3	EID	PID	G1	G2	G3	ZOFFS	THETA or MCID		
		TFLAG	T1	T2	T3				

```
CQUAD4   1       1       1       2       13      12
CQUAD4   2       1       2       3       14      13
CTRIA3   41      1       64      22      11
```

Figure 66 NASTRAN Card Image for Quadrilateral and Triangular Linear Elements

The solver attributes up to the THETA field are familiar (element ID, property ID, Grid ID connectivity list). The THETA field defines orientation of an element material direction, ZOFFS defines an offset of the plane of the element along the element normal direction, TFLAG and T(i) are used to define membrane thicknesses at the grid points for tapered elements. None of these additional fields are used in the pilot study but the material orientation is commonly used and therefore discussed below.

The THETA field can contain either an angle or a coordinate system ID that orients the element material coordinate system. If an angle is defined, the material coordinate system is rotated by this angle from the element coordinate system about the element normal. If a coordinate system ID is specified, the X-axis of that system is projected onto the plane of the element which defines a vector direction relative to the element coordinate system x-axis. If blank, the element material coordinate system is the element coordinate system. This is especially important when composite properties or any orthotropic or anisotropic materials are used. The AP209 ed2 data model supports specification of these types of element material coordinate systems; however, the ATS3 pilot model does not define a separate material coordinate direction. Therefore, a parametric_surface_3d_element_coordinate_system instance as shown in Figure 67 is shared for each element type.

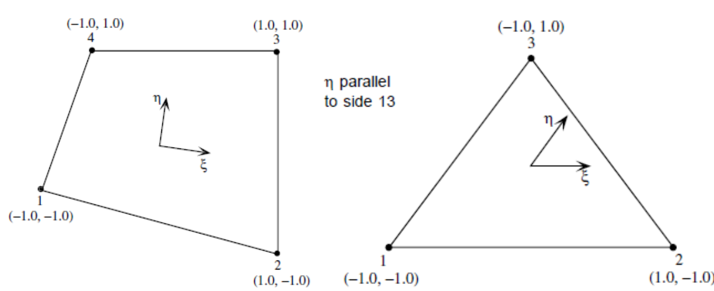


Figure 67 Parametric Element Coordinate Systems for Quadrilateral and Triangular Elements

The fundamental concept is that the coordinate systems are parametric and can be computed using the geometric properties of the element instance. The details of element coordinate system development are described in the NASTRAN quick reference manual or theoretical manuals but is beyond the scope of this document. The Part 104 document discusses the selection of element coordinate systems in great detail in Section 5.8 Element topologies and in Section 5.9 Element coordinate systems. A thorough review of these sections is recommended.

7.5.3 STEP Entities

The STEP file listing of the elements and related entities for the ATS3 model are shown in Figure 68, and the EXPRESS schema definitions for these entities are shown in Figure 69 and Figure 70.

```
#454= SURFACE_3D_ELEMENT_REPRESENTATION('1',(#463),#455,(#80,#89,#133,#129),#67,#468,#465,#423);
#473= SURFACE_3D_ELEMENT_REPRESENTATION('2',(#463),#455,(#89,#93,#137,#133),#67,#468,#465,#423);
#590= SURFACE_3D_ELEMENT_REPRESENTATION('41',(#592),#455,(#329,#169,#125),#67,#594,#465,#423);

#468= SURFACE_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_QUADRILATERAL.CQUAD4',(
(ENUMERATED_SURFACE_ELEMENT_PURPOSE(.MEMBRANE_DIRECT.),
 ENUMERATED_SURFACE_ELEMENT_PURPOSE(.MEMBRANE_SHEAR.)),
(ENUMERATED_SURFACE_ELEMENT_PURPOSE(.BENDING_DIRECT.),
 ENUMERATED_SURFACE_ELEMENT_PURPOSE(.BENDING_TORSION.)),
(ENUMERATED_SURFACE_ELEMENT_PURPOSE(.NORMAL_TO_PLANE_SHEAR.))
),.QUADRILATERAL.);
#594= SURFACE_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_TRIANGLE.CTRIA3',(
(ENUMERATED_SURFACE_ELEMENT_PURPOSE(.MEMBRANE_DIRECT.),
 ENUMERATED_SURFACE_ELEMENT_PURPOSE(.MEMBRANE_SHEAR.)),
(ENUMERATED_SURFACE_ELEMENT_PURPOSE(.BENDING_DIRECT.),
 ENUMERATED_SURFACE_ELEMENT_PURPOSE(.BENDING_TORSION.)),
(ENUMERATED_SURFACE_ELEMENT_PURPOSE(.NORMAL_TO_PLANE_SHEAR.))
),.TRIANGLE.);

#463= PARAMETRIC_SURFACE_3D_ELEMENT_COORDINATE_SYSTEM('',1,0.);
#592= PARAMETRIC_SURFACE_3D_ELEMENT_COORDINATE_SYSTEM('',1,0.);

#455= /* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CONTEXT()
REPRESENTATION_CONTEXT('REP_CONTEXT_FEA','3D'));
```

Figure 68 STEP File Fragment for Surface 3D Elements and Related Entities

```
ENTITY Element_Descriptor
    SUPERTYPE OF (
        ONEOF (
            Volume_3d_Element_Descriptor,
            Axisymmetric_Volume_2d_Element_Descriptor,
            Plane_Volume_2d_Element_Descriptor,
            Surface_3d_Element_Descriptor,
            Axisymmetric_Surface_2d_Element_Descriptor,
            Plane_Surface_2d_Element_Descriptor,
            Curve_3d_Element_Descriptor,
            Axisymmetric_Curve_2d_Element_Descriptor,
            Plane_Curve_2d_Element_Descriptor ) );
      topology_order  : Element_Order;
      description     : Text;
  END_ENTITY;

  ENTITY Surface_3d_Element_Descriptor
    SUBTYPE OF ( Element_Descriptor );
      purpose  : SET [1 : ?] OF SET [1 : ?] OF Surface_Element_Purpose;
      shape    : Element_2d_Shape;
  END_ENTITY;
```

Figure 69 EXPRESS Schema for Surface 3D Element Descriptor

The schema listing in Figure 69 has not been flattened and the supertype-subtype relationships are explicitly defined in separate entity definitions that must be navigated to understand the full definition of a specialized class such as surface_3d_element_descriptor. Both styles of schema have been presented in this document to familiarize the reader with the alternate formats. The SMRL uses the supertype-subtype format.

Additional type information related to the element descriptors is used to populate the attributes. This information defines many behavior aspects of the elements as appropriate for the particular solver used as the data source or the intended target solver format.

```
TYPE Element_Order = ENUMERATION OF
  ( Linear_Order,
   Quadratic_Order,
   Cubic_Order );
END_TYPE;

TYPE Surface_Element_Purpose = SELECT
  ( Enumerated_Surface_Element_Purpose,
   Application_Defined_Element_Purpose );
END_TYPE;

TYPE Enumerated_Surface_Element_Purpose = ENUMERATION OF
  ( Membrane_Direct,
   Membrane_Shear,
   Bending_Direct,
   Bending_Torsion,
   Normal_To_Plane_Shear );
END_TYPE;

TYPE Element_2d_Shape = ENUMERATION OF
  ( Quadrilateral,
   Triangle );
END_TYPE;
```

Figure 70 EXPRESS Schema Definition for Surface Element 3D Element Descriptor Attributes

Use of element descriptors for describing element behavior in STEP is a mechanism to provide enumerated terms that can be used to recreate some (but not all) of the mathematics used in the element formulation (not the actual mathematical formulations). As an example, the NASTRAN CQUAD4 and TRIA3 elements have fundamental differences in element formulations that effect the variation of strain over the surface of the element. However, the descriptors for each enumerate similar capabilities and behaviors.

To capture these mathematical differences, a lower level construct in AP209 ed2 can be used to describe the element matrix integration schemes involved in the element formulation. Section 5.10 Element matrix integration of Part 104 details these aspects of element formulation and is the starting point for further investigation of this topic. However, this is an advanced topic that is not covered in any more detail for the pilot study.

The full EXPRESS schema definition of the surface_3d_element_representation entity is complex and makes extensive use of WHERE rules in the schema definition to ensure referential integrity of the AP209 ed2 element data model. Figure 71 lists the specialized surface_3d_element_representation and its immediate supertype, element_representation.

```
ENTITY Element_Representation
  SUPERTYPE OF (
      ONEOF (
          Volume_3d_Element_Representation,
          Axisymmetric_Volume_2d_Element_Representation,
          Plane_Volume_2d_Element_Representation,
          Surface_3d_Element_Representation,
          Axisymmetric_Surface_2d_Element_Representation,
          Plane_Surface_2d_Element_Representation,
          Curve_3d_Element_Representation,
          Axisymmetric_Curve_2d_Element_Representation,
          Plane_Curve_2d_Element_Representation,
          Point_Element_Representation,
          Directionally_Explicit_Element_Representation,
          Explicit_Element_Representation,
          Substructure_Element_Representation ) )
  SUBTYPE OF ( Representation );
    node_list  : LIST [1 : ?] OF Node_Representation;
  WHERE
    wr1:( SIZEOF( QUERY ( item <* node_list | ( ( 'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'GEOMETRIC_NODE' )
          IN TYPEOF( item ) ) ) ) = 0 );
END_ENTITY;

ENTITY Surface_3d_Element_Representation
  SUBTYPE OF ( Element_Representation );
    model_ref           : Fea_Model_3d;
    element_descriptor  : Surface_3d_Element_Descriptor;
    property            : Surface_Element_Property;
    material            : Element_Material;
  UNIQUE
    ur1 : model_ref, SELF\Representation.name;
  WHERE
    wr1: ( ( SIZEOF( QUERY ( item <* SELF\Representation.items | ( ( 'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' +
          'PARAMETRIC_SURFACE_3D_ELEMENT_COORDINATE_SYSTEM'  ) IN TYPEOF( item ) ) ) ) ) + SIZEOF( QUERY ( item <* SELF\
          Representation.items | ( ( 'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' +
          'CONSTANT_SURFACE_3D_ELEMENT_COORDINATE_SYSTEM'  ) IN TYPEOF( item ) ) ) ) ) + SIZEOF( QUERY ( item <* SELF\
          Representation.items | ( ( 'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' +
          'ALIGNED_SURFACE_3D_ELEMENT_COORDINATE_SYSTEM'  ) IN TYPEOF( item ) ) ) ) ) = 1 );
    wr2: ( SIZEOF( QUERY ( item1 <* material.properties | ( SIZEOF( QUERY ( item2 <*
item1\Property_Definition_Representation.
        used_representation.items | ( SIZEOF( [
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_LINEAR_ELASTICITY' ,
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_MASS_DENSITY',
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_AREA_DENSITY',
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_TANGENTIAL_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION',
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_SECANT_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION',
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_MOISTURE_ABSORPTION',
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_SHELL_MEMBRANE_STIFFNESS',
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_SHELL_BENDING_STIFFNESS',
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_SHELL_MEMBRANE_BENDING_COUPLING_STIFFNESS',
        'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.' + 'FEA_SHELL_SHEAR_STIFFNESS'] * TYPEOF( item2 ) ) = 1 ) ) )
          = 1 ) ) ) >= 1 );
    wr3: ( 'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF.PARAMETRIC_REPRESENTATION_CONTEXT' IN TYPEOF( SELF\
          Representation.context_of_items ) );
    fu1: Required_2d_Nodes( SELF\Element_Representation.node_list, Element_Descriptor.shape, Element_Descriptor\
          Element_Descriptor.topology_order );
END_ENTITY;
```

Figure 71 EXPRESS Schema Definition for Surface_3D_Element_Representation

7.6 Property Definition

The property definitions for surface 3D elements are not as complex as the 1D elements but require some explanation. For the ATS3 pilot model, a single NASTRAN property is used that specifies a uniform thickness value normal to the 2-dimensional plane of each element. This thickness provides the missing 3rd dimension for surface based shell element models of 3 dimensional volumes.

This is an approximation to a full 3D volume model, but it provides an efficient mechanism to reduce the size of the analysis models by eliminating the redundant thickness dimension from the geometrical definition of the FEM. The accuracy of a FEM solution using this approximation is dependent upon many factors, one of which is that the length and width of the element should be much greater than the thickness. This 'thin shell' assumption is not strictly followed in this pilot model, however, the effect is on the values produced by the solution process, not the representation of the model and solution in a AP209 ed2 data model. In other words, the reader is cautioned that bad solution data can be represented in the STEP data model as easily as any other data format.

7.6.1 STEP Diagram

The instance diagram for the surface_element_property is no more complex than the instances shown in Figure 64. All element instances are included in Figure 72 to illustrate the shared property usage.

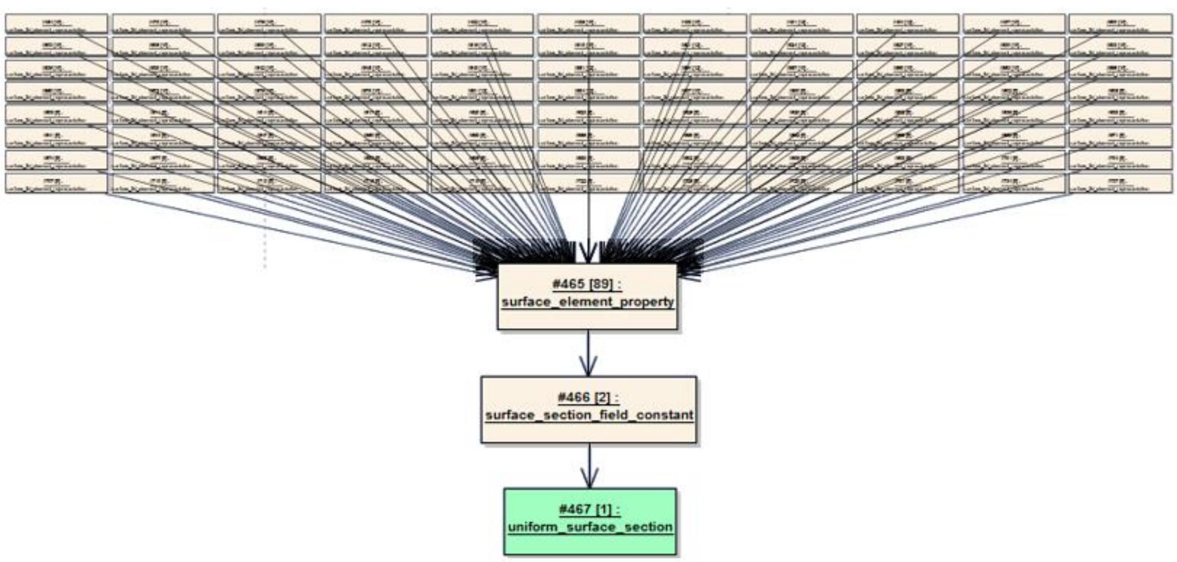


Figure 72 STEP Instance Diagram Showing Single Shared Surface_Element_Property

The uniform_surface_section and surface_section_field_constant entities found in this diagram represent a constant thickness value over the surface of the element. Non-uniform section information (such as tapered thickness over the element) can be represented by specifying a list of uniform_surface_section entities in a surface_section_field_varying entity. The reader is referred to the recommended practice document and the schema definitions in the SMRL. Additionally, other properties related to the behavior of the through-the-thickness section are provided and can be used to represent further the behavior of the element. If any of these additional properties is supplied by an individual element (such as element offset), unique field and section definitions must be created and related to a unique surface_element_property.

7.6.2 FE Model Input

The single NASTRAN property card for the ATS3 model is shown in Figure 73. Only the material and thickness attributes are used in the pilot study models.

1	2	3	4	5	6	7	8	9	10
PSHELL	PID	MID1	T	MID2	12I/T**3	MID3	TS/T	NSM	
	Z1	Z2	MID4						

```
PSHELL   1         1         2.        1                   1
```

Figure 73 NASTRAN Card Input for PSHELL Property

7.6.3 STEP Entities

The STEP instances shown in Figure 74 represent the shell property for the ATS3 model. The uniform_surface_section listing has been reformatted for easy comparison with the EXPRESS schema figure for these entities.

```
#465= SURFACE_ELEMENT_PROPERTY('PSHELL.1','',#466);

#466= SURFACE_SECTION_FIELD_CONSTANT(#467);
#467= UNIFORM_SURFACE_SECTION(
       UNSPECIFIED_VALUE(.UNSPECIFIED.),
       UNSPECIFIED_VALUE(.UNSPECIFIED.),
       UNSPECIFIED_VALUE(.UNSPECIFIED.),
       2.,
       UNSPECIFIED_VALUE(.UNSPECIFIED.),
       UNSPECIFIED_VALUE(.UNSPECIFIED.));
```

Figure 74 STEP File Fragment for Surface_Element_Property and Related Entities

The EXPRESS schema definitions for these entities and the supertypes are shown in Figure 75.

```
ENTITY Surface_Element_Property;
    property_id  : Identifier;
    description  : Text;
    section      : Surface_Section_Field;
END_ENTITY;

ENTITY Surface_Section_Field
  SUPERTYPE OF (
     ONEOF (
         Surface_Section_Field_Constant,
         Surface_Section_Field_Varying ) );
END_ENTITY;

ENTITY Surface_Section_Field_Constant
  SUBTYPE OF ( Surface_Section_Field );
    definition  : Surface_Section;
END_ENTITY;

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;

ENTITY Uniform_Surface_Section
  SUBTYPE OF ( Surface_Section );
    thickness          : Context_Dependent_Measure;
    bending_thickness  : Measure_Or_Unspecified_Value;
    shear_thickness    : Measure_Or_Unspecified_Value;
END_ENTITY;
```

Figure 75 EXPRESS Schema for Surface_Element_Property and Related Entities

7.7 Material Definition

The material definition and discussion are identical to that in Section 5.7.

7.7.1 STEP Diagram

Identical to Section 5.7.1.

7.7.2 FE Model Input

Identical to Section 5.7.2.

7.7.3 STEP Entities

Identical to Section 5.7.3.

8 Pilot Model ATS4 – 3D Solid Element

The ATS4pilot model composition is illustrated in Figure 76.

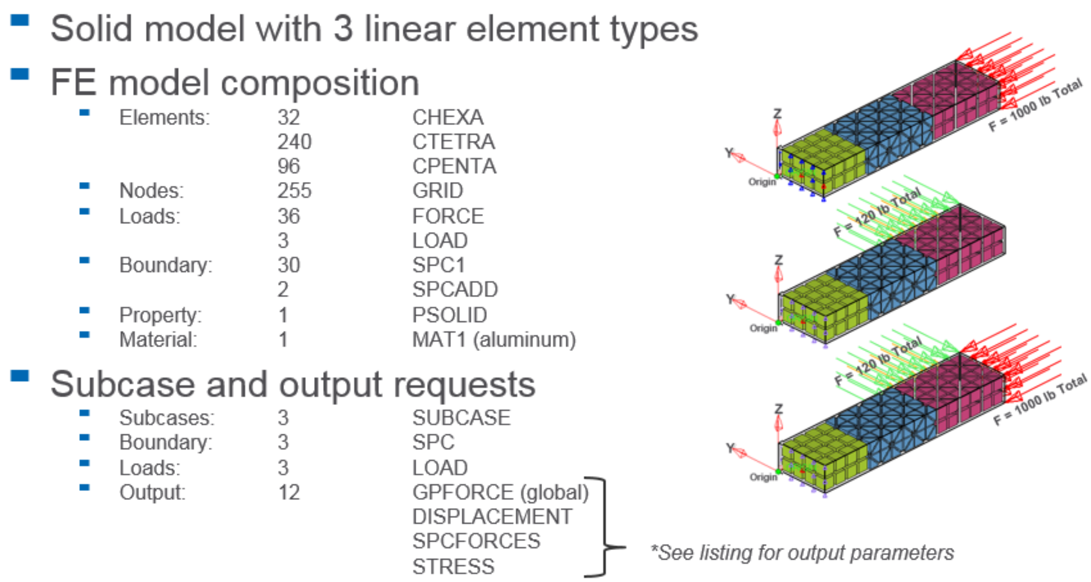


Figure 76 Pilot Model Content Summary ATS4

8.1 Product Definition Shape

The structure of the product definition entity graph is similar to the 2D shell element ATS3 pilot model; however, the instance identifiers and association counts shown in Figure 77 are different.

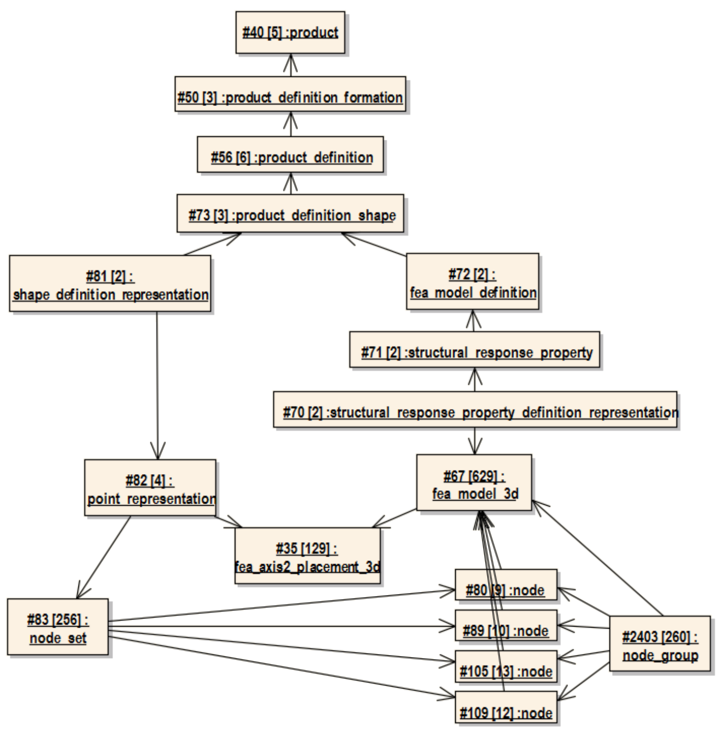


Figure 77 ATS4 FEM High Level Product Definition Shape Structure

8.2 FEA Model

The FEA model for the ATS4 pilot model is similar to the ATS3 pilot model. No local coordinate systems are present. All geometric founding is with respect to the basic coordinate system.

8.3 Coordinate System

No local coordinate systems are defined for the ATS4 model. Only the basic coordinate system is used.

8.3.1 STEP Diagram

No additional diagram is needed for coordinate systems.

8.3.2 FE Model Input

The NASTRAN basic coordinate system is an implied system; no user input is required.

8.3.3 STEP Entities

No entities beyond the fea_model_3d, point_representation and fea_axis2_placement_3d shown in Figure 77 are required to describe the basic coordinate system and the geometric founding of this model.

8.4 Node Definition

The ATS4 pilot study model uses 3-dimensional solid elements which require a 3-dimensional grid of node locations to define the discretized analysis space. The nodes are arranged in 3 layers at z=0, 1 and 2 inches. Each of these node layers is laid out as a 5 x 17 array of equally spaces nodes for a total of 255 nodes.

8.4.1 STEP Diagram

The details of the node definitions for ththe ATS4 model are similar to the prior 2D shell ATS3 model. Refer to Section 7.4.1 for details.

8.4.2 FE Model Input

Refer to Section 7.4.2 for input formats. The complete listing of the NASTRAN grid point inputs is included in the Appendix of this document.

8.4.3 STEP Entities

The instances for the first 5 node entities is shown in Figure 78.

```
#80= NODE('1',(#78),#36,#67);
#78= CARTESIAN_POINT('1',(0.,-4.,-3.431520000000000E-8));
#89= NODE('2',(#87),#36,#67);
#87= CARTESIAN_POINT('2',(0.,-3.,-3.431520000000000E-8));
#93= NODE('3',(#91),#36,#67);
#91= CARTESIAN_POINT('3',(0.,-2.,-3.431520000000000E-8));
#97= NODE('4',(#95),#36,#67);
#95= CARTESIAN_POINT('4',(0.,-1.,-3.431520000000000E-8));
#101= NODE('5',(#99),#36,#67);
#99= CARTESIAN_POINT('5',(0.,-6.013760000000000E-9,-3.431520000000000E-8));

#83= NODE_SET('node_set_name_1',(#80,#89,#93,#97,#101,#105,#109,#113,
#117,#121,#125,#129,#133,#137,#141,#145,#149,#153,#157,#161,#165,#169,
…removed for clarity…
,#1013,#1017,#1021,#1025,#1029,#1033,#1037,#1041,#1045,#1049,#1053,#1057
,#1061,#1065,#1069,#1073,#1077,#1081,#1085,#1089,#1093,#1097,#1101));
```

Figure 78 STEP File Fragment for AT43 Nodes

8.5 Element Definition

The ATS4 pilot model introduces the solid volumetric element type to the study. These elements fully define the analysis domain of the element and require a 3-dimensional grid of node coordinates to specify the connectivity of the elements. The elements are 4-noded tetrahedron, 6-noded wedge and 8-noded hexahedron elements.

Special care must be given to proper mapping of the connectivity node list. The solver specification for connectivity (NASTRAN) may use a different node order than the STEP AP209 ed2 standard, depending on the mathematical order of the element (linear, quadratic, cubic, etc..). For this pilot study, linear elements were selected and the node order for the element vertices is consistent. However, this is an exception and not a rule. Figure 79 shows the node order definitions for a quadratic hexahedron element. Note the different mid-side node IDs and the lack of a centroid and face center nodes in the NASTRAN definitions. Refer to the Part 104 document for complete topology (edge and face) definitions for all supported elements shapes and orders. However, as stated earlier, the linear element uses the circled vertex nodes that are numbered similarly.

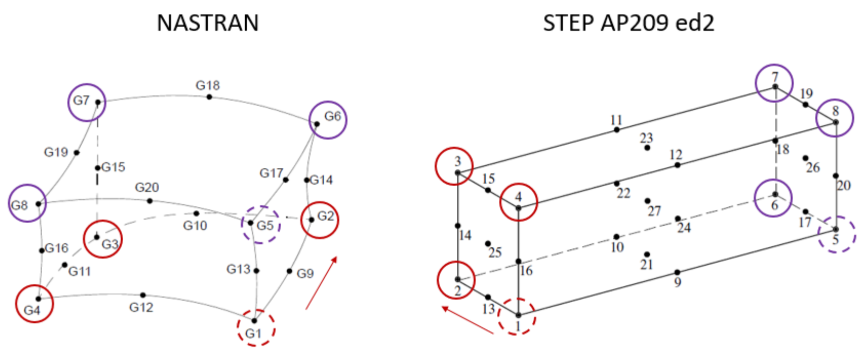


Figure 79 Node Order Mapping for Quadratic Hexahedron Element Shape

The solver format for volumetric elements also supports additional input that specifies the internal formulations of the element in the solver. These options are not included in this initial pilot study.

8.5.1 STEP Diagram

The STEP instance diagram illustrating the element connectivity and associated entities is shown in Figure 80. In the ATS4 model, very few entities besides the element instance are required to define the FEM element. The additional entities are discussed in the following sections.

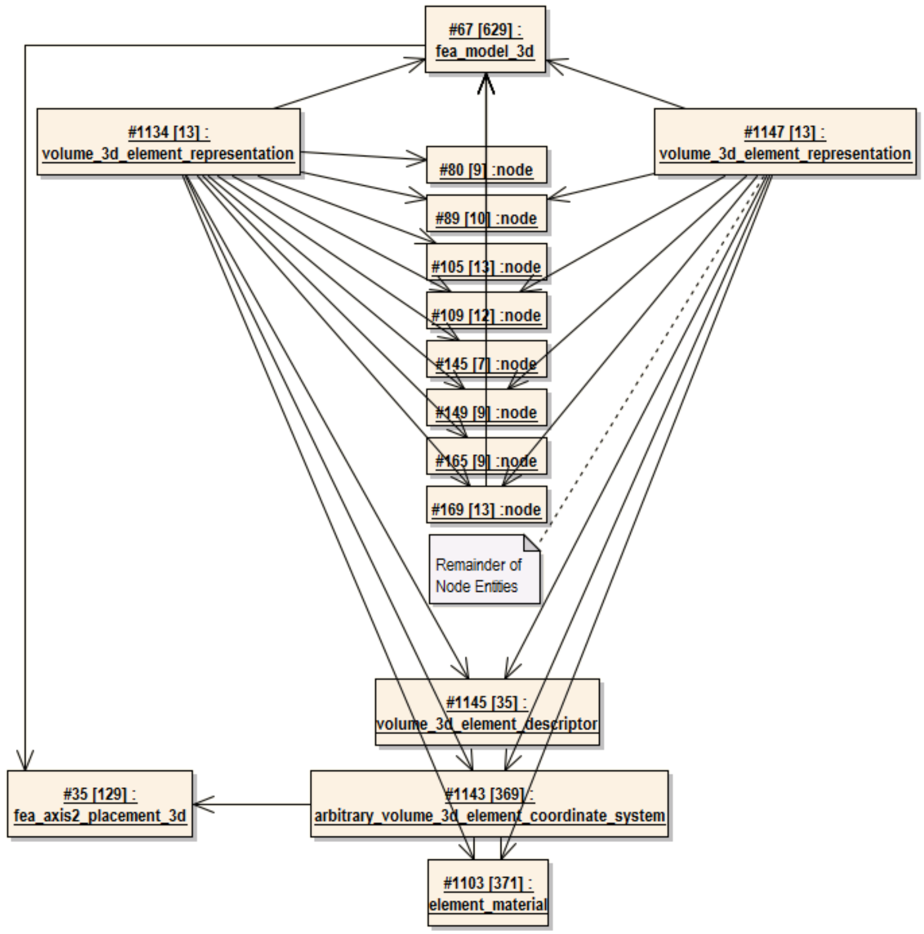


Figure 80 STEP Volume 3D Element Definition and Related Data Model Structures

All the nodes for the first volume_3d_element_representation are shown in the diagram. The second element shares 4 nodes that typically indicates adjacent solid elements that share a common face. Similar structures are found in the STEP data model for each volume_3d_element_descriptor instance. In other hybrid models (volume, surface and curve based elements in the same model), there will be combinations of all the structures discussed in all of the ATS models.

Additionally, the particular NASTRAN definition for the solid element property leads to the selection of the arbitrary_volume_3d_element_coordinate_system. This will be discussed in Section 8.6.

Lastly, the entire ATS4 model uses the same isotropic material definition discussed in Section 5.7.

8.5.2 FE Model Input

The solver format for specifying the linear hexahedron volumetric element in NASTRAN is shown in Figure 81.

1	2	3	4	5	6	7	8	9	10
CHEXA	EID	PID	G1	G2	G3	G4	G5	G6	
	G7	G8	G9	G10	G11	G12	G13	G14	
	G15	G16	G17	G18	G19	G20			

```
CHEXA   1       1       1       2       7       6       16      17
        22      21
CHEXA   2       1       2       3       8       7       17      18
        23      22
CHEXA   3       1       3       4       9       8       18      19
        24      23
```

Figure 81 NASTRAN Card Image for Linear Hexahedron Elements

The formats for the tetraheadon and wedge solid elements are described in the NASTRAN reference manual.

8.5.3 STEP Entities

A brief listing of the Part 21 text is shown in Figure 82. Only hexahedron elements are listed but all the element descriptor are presented.

```
#1135=/* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CONTEXT()
REPRESENTATION_CONTEXT('REP_CONTEXT_FEA','3D'));

#1134= VOLUME_3D_ELEMENT_REPRESENTATION('1',(#1143),#1135,(#80,#89,#109,
#105,#145,#149,#169,#165),#67,#1145,#1103);
#1147= VOLUME_3D_ELEMENT_REPRESENTATION('2',(#1143),#1135,(#89,#93,#113,
#109,#149,#153,#173,#169),#67,#1145,#1103);
#1150= VOLUME_3D_ELEMENT_REPRESENTATION('3',(#1143),#1135,(#93,#97,#117,
#113,#153,#157,#177,#173),#67,#1145,#1103);

#1145= VOLUME_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_HEXAHEDRON.CHEXA',
(ENUMERATED_VOLUME_ELEMENT_PURPOSE(.STRESS_DISPLACEMENT.)),.HEXAHEDRON.);
#1243= VOLUME_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_TETRAHEDRON.CTETRA',
(ENUMERATED_VOLUME_ELEMENT_PURPOSE(.STRESS_DISPLACEMENT.)),.TETRAHEDRON.);
#1965= VOLUME_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_WEDGE.CPENTA',
(ENUMERATED_VOLUME_ELEMENT_PURPOSE(.STRESS_DISPLACEMENT.)),.WEDGE.);
```

Figure 82 STEP File Fragment for Volume 3D Element Information

8.6 Property Definition

The property definition for a solid element is minimal; however its associations to an element material and a local element coordinate system are required.

8.6.1 STEP Diagram

The diagram in Figure 80 shows the relationship of volumetric elements to the element material and local element coordinate system.

8.6.2 FE Model Input

The NASTRAN PSOLID card shown in Figure 83 is used to associate both the element material and a coordinate system to the related element; it is shared between all element types in this instance.

1	2	3	4	5	6	7	8	9	10
PSOLID	PID	MID	CORDM	IN	STRESS	ISOP	FCTN	COROT	

```
PSOLID   1         1         0
```

Figure 83 NASTRAN Volume Element Property Card

The specification in NASTRAN of the CORDM=0 is an explicit reference to a material coordinate system for the element. This maps to the arbitrary_volume_3d_element_coordinate_system entity shown in Figure 80 which in-turn refers to the basic coordinate system. The other options on the PSOLID card are not used in this pilot model.

8.6.3 STEP Entities

No additional STEP entities are associated with the volumetric property. Figure 80 in Section 8.5.1 fully defines the required AP209 ed2 instances.

8.7 Material Definition

The material definition and discussion is identical to that presented in Section 5.7.

8.7.1 STEP Diagram

Identical to Section 5.7.1.

8.7.2 FE Model Input

Identical to Section 5.7.2.

8.7.3 STEP Entities

Identical to Section 5.7.3.

9 References and Links

The following are a list of useful links which the reader can use to learn more about the STEP standard, the US and European organizations that administer, develop and support these standards, and the LOTAR Engineering and Simulation work group, which authored this handbook.

The reader is encouraged to visit all the links for an introduction and overview of the activities of these organizations. However, the LOTAR EAS WG link and the CAx-IF and CAE-IF links contain reference material that is directly applicatble to the topics discussed in the handbook.

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
CAx-IF	https://www.cax-if.org
CAE-IF	http://afnet.fr/dotank/sps/cae-if

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

10 Appendix A – NASTRAN Input File Full Listing

The NASTRAN input files for the 4 pilot models, as described in the following 4 subsections, were used for this pilot study. They are not expected to be changed or updated unless a significant issue is identified that materially affects the generation of the STEP files. The first load case is defined consistently for all models with appropriate loads and boundary conditions to enable comparable solution data. However, the results may not be numerically identical due to differences in the model abstractions and element selection.

10.1 ATS1m4.bdf

```
$ Linear Static Analysis
SOL 101
TIME 600
CEND
$
SEALL = ALL
SUPER = ALL
TITLE = Nastran job EAS test case ATS1m4
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
$
BEGIN BULK
PARAM   POST     -1
PARAM   AUTOSPC  YES
PARAM   NOCOMPS  -1
PARAM   PRTMAXIM YES
$
PROD     1       1       8.      0.
$
CROD     1       1       1       2
CROD     2       1       2       3
CROD     3       1       3       4
CROD     4       1       4       5
CROD     5       1       5       6
CROD     6       1       6       7
CROD     7       1       7       8
CROD     8       1       8       9
CROD     9       1       9       10
CROD     10      1       10      11
CROD     11      1       11      12
CROD     12      1       12      13
CROD     13      1       13      14
CROD     14      1       14      15
CROD     15      1       15      16
CROD     16      1       16      17
$
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.
$
SPC1     100     123     1
$
FORCE    200     17      0       1000.   -1.      0.      0.
$
CORD2R   1               0.      0.      0.      0.      0.      1.
         1.      0.      0.
$
ENDDATA
```

10.2 ATS2m4.bdf

```
$ Linear Static Analysis
SOL 101
TIME 600
CEND
$
SEALL = ALL
SUPER = ALL
TITLE = Nastran EAS test case ATS2m4
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
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
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
$
BEGIN BULK
PARAM   POST    -1
PARAM   AUTOSPC YES
PARAM   NOCOMPS -1
PARAM   PRTMAXIM YES
$
PBAR    1       1       8.      2.667   10.667  0.      0.
        1.      2.      1.      -2.     0.      0.      0.      0.
                        0.
$
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
```

10.3 ATS3m4.bdf

```
$ Linear Static Analysis
SOL 101
TIME 600
CEND
$
SEALL = ALL
SUPER = ALL
TITLE = Nastran job EAS test case ATS3m4
ECHO = NONE
MAXLINES = 999999999
GPFORCE(PUNCH) = ALL
$
SUBCASE 1
   SUBTITLE=subcase1
   SPC = 11
   LOAD = 21
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
SUBCASE 2
   SUBTITLE=subcase2
   SPC = 12
   LOAD = 22
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
SUBCASE 3
   SUBTITLE=subcase3
   SPC = 12
   LOAD = 23
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
SUBCASE 4
   SUBTITLE=subcase4
   SPC = 103
   LOAD = 500
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
$
BEGIN BULK
PARAM   POST     -1
PARAM   AUTOSPC  YES
PARAM   NOCOMPS  -1
PARAM   PRTMAXIM YES
$
PSHELL  1       1       2.      1               1
$
CQUAD4  1       1       1       2       13      12
CQUAD4  2       1       2       3       14      13
CQUAD4  3       1       3       4       15      14
CQUAD4  4       1       4       5       16      15
CQUAD4  5       1       5       6       17      16
CQUAD4  6       1       6       7       18      17
CQUAD4  7       1       7       8       19      18
CQUAD4  8       1       8       9       20      19
CQUAD4  9       1       9       10      21      20
CQUAD4  10      1       10      11      22      21
CQUAD4  11      1       12      13      24      23
CQUAD4  12      1       13      14      25      24
CQUAD4  13      1       14      15      26      25
CQUAD4  14      1       15      16      27      26
CQUAD4  15      1       16      17      28      27
CQUAD4  16      1       17      18      29      28
CQUAD4  17      1       18      19      30      29
CQUAD4  18      1       19      20      31      30
CQUAD4  19      1       20      21      32      31
CQUAD4  20      1       21      22      33      32
```

```
CQUAD4   21      1       23      24      35      34
CQUAD4   22      1       24      25      36      35
CQUAD4   23      1       25      26      37      36
CQUAD4   24      1       26      27      38      37
CQUAD4   25      1       27      28      39      38
CQUAD4   26      1       28      29      40      39
CQUAD4   27      1       29      30      41      40
CQUAD4   28      1       30      31      42      41
CQUAD4   29      1       31      32      43      42
CQUAD4   30      1       32      33      44      43
CQUAD4   31      1       34      35      46      45
CQUAD4   32      1       35      36      47      46
CQUAD4   33      1       36      37      48      47
CQUAD4   34      1       37      38      49      48
CQUAD4   35      1       38      39      50      49
CQUAD4   36      1       39      40      51      50
CQUAD4   37      1       40      41      52      51
CQUAD4   38      1       41      42      53      52
CQUAD4   39      1       42      43      54      53
CQUAD4   40      1       43      44      55      54
CTRIA3   41      1       64      22      11
CTRIA3   42      1       11      57      64
CTRIA3   43      1       57      58      64
CTRIA3   44      1       65      64      58
CTRIA3   45      1       66      65      58
CTRIA3   46      1       58      59      66
CTRIA3   47      1       59      60      66
CTRIA3   48      1       67      66      60
CTRIA3   49      1       68      67      60
CTRIA3   50      1       60      61      68
CTRIA3   51      1       61      62      68
CTRIA3   52      1       69      68      62
CTRIA3   53      1       22      64      33
CTRIA3   54      1       71      33      64
CTRIA3   55      1       72      71      64
CTRIA3   56      1       64      65      72
CTRIA3   57      1       65      66      72
CTRIA3   58      1       73      72      66
CTRIA3   59      1       74      73      66
CTRIA3   60      1       66      67      74
CTRIA3   61      1       67      68      74
CTRIA3   62      1       75      74      68
CTRIA3   63      1       76      75      68
CTRIA3   64      1       68      69      76
CTRIA3   65      1       78      44      33
CTRIA3   66      1       33      71      78
CTRIA3   67      1       71      72      78
CTRIA3   68      1       79      78      72
CTRIA3   69      1       80      79      72
CTRIA3   70      1       72      73      80
CTRIA3   71      1       73      74      80
CTRIA3   72      1       81      80      74
CTRIA3   73      1       82      81      74
CTRIA3   74      1       74      75      82
CTRIA3   75      1       75      76      82
CTRIA3   76      1       83      82      76
CTRIA3   77      1       44      78      55
CTRIA3   78      1       85      55      78
CTRIA3   79      1       86      85      78
CTRIA3   80      1       78      79      86
CTRIA3   81      1       79      80      86
CTRIA3   82      1       87      86      80
CTRIA3   83      1       88      87      80
CTRIA3   84      1       80      81      88
CTRIA3   85      1       81      82      88
CTRIA3   86      1       89      88      82
CTRIA3   87      1       90      89      82
CTRIA3   88      1       82      83      90
$
MAT1     1       1.+7            .33     2.54-4  1.3-5   70.
$
```

```
GRID*   1                          -6.24022-8    -4.
*       1.
GRID    2             1.    -4.     1.
GRID    3             2.    -4.     1.
GRID    4             3.    -4.     1.
GRID    5             4.    -4.     1.
GRID    6             5.    -4.     1.
GRID    7             6.    -4.     1.
GRID    8             7.    -4.     1.
GRID    9             8.    -4.     1.
GRID    10            9.    -4.     1.
GRID    11            10.   -4.     1.
GRID*   12                         -6.24022-8    -3.
*       1.
GRID    13            1.    -3.     1.
GRID    14            2.    -3.     1.
GRID    15            3.    -3.     1.
GRID    16            4.    -3.     1.
GRID    17            5.    -3.     1.
GRID    18            6.    -3.     1.
GRID    19            7.    -3.     1.
GRID    20            8.    -3.     1.
GRID    21            9.    -3.     1.
GRID    22            10.   -3.     1.
GRID*   23                         -6.24022-8    -2.
*       1.
GRID    24            1.    -2.     1.
GRID    25            2.    -2.     1.
GRID    26            3.    -2.     1.
GRID    27            4.    -2.     1.
GRID    28            5.    -2.     1.
GRID    29            6.    -2.     1.
GRID    30            7.    -2.     1.
GRID    31            8.    -2.     1.
GRID    32            9.    -2.     1.
GRID    33            10.   -2.     1.
GRID*   34                         -6.24022-8    -1.
*       1.
GRID    35            1.    -1.     1.
GRID    36            2.    -1.     1.
GRID    37            3.    -1.     1.
GRID    38            4.    -1.     1.
GRID    39            5.    -1.     1.
GRID    40            6.    -1.     1.
GRID    41            7.    -1.     1.
GRID    42            8.    -1.     1.
GRID    43            9.    -1.     1.
GRID    44            10.   -1.     1.
GRID*   45                         -6.24022-8    -5.27577-8
*       1.
GRID*   46                          1.           -5.27577-8
*       1.
GRID*   47                          2.           -5.27577-8
*       1.
GRID*   48                          3.           -5.27577-8
*       1.
GRID*   49                          4.           -5.27577-8
*       1.
GRID*   50                          5.           -5.27577-8
*       1.
GRID*   51                          6.           -5.27577-8
*       1.
GRID*   52                          7.           -5.27577-8
*       1.
GRID*   53                          8.           -5.27577-8
*       1.
GRID*   54                          9.           -5.27577-8
*       1.
GRID*   55                          10.          -5.27577-8
*       1.
GRID    57            11.   -4.     1.
```

```
GRID     58              12.     -4.     1.
GRID     59              13.     -4.     1.
GRID     60              14.     -4.     1.
GRID     61              15.     -4.     1.
GRID     62              16.     -4.     1.
GRID     64              11.     -3.     1.
GRID     65              12.     -3.     1.
GRID     66              13.     -3.     1.
GRID     67              14.     -3.     1.
GRID     68              15.     -3.     1.
GRID     69              16.     -3.     1.
GRID     71              11.     -2.     1.
GRID     72              12.     -2.     1.
GRID     73              13.     -2.     1.
GRID     74              14.     -2.     1.
GRID     75              15.     -2.     1.
GRID     76              16.     -2.     1.
GRID     78              11.     -1.     1.
GRID     79              12.     -1.     1.
GRID     80              13.     -1.     1.
GRID     81              14.     -1.     1.
GRID     82              15.     -1.     1.
GRID     83              16.     -1.     1.
GRID*    85                              11.             -5.27577-8
*        1.
GRID*    86                              12.             -5.27577-8
*        1.
GRID*    87                              13.             -5.27577-8
*        1.
GRID*    88                              14.             -5.27577-8
*        1.
GRID*    89                              15.             -5.27577-8
*        1.
GRID*    90                              16.             -5.27577-8
*        1.
$
SPCADD   11      100     101     110
SPCADD   12      100     102     110
$
LOAD     21      1.      1.      200
LOAD     22      1.      1.      300     1.      400
LOAD     23      1.      1.      200     1.      300     1.      400
$
SPC1     100     123     23
SPC1     101     1       1       12      34      45
SPC1     102     123     1       12      34      45
SPC1     103     123     23
SPC1     103     23      76
SPC1     103     3       1       12      34      45
SPC1     103     3       62      69      83      90
SPC1     110     45      1       THRU    55
SPC1     110     45      57      58      59      60      61      62
         64      65      66      67      68      69      71      72
         73      74      75      76      78      79      80      81
         82      83      85      86      87      88      89      90
$
FORCE    200     62      0       125.    -1.     0.      0.
FORCE    200     69      0       250.    -1.     0.      0.
FORCE    200     76      0       250.    -1.     0.      0.
FORCE    200     83      0       250.    -1.     0.      0.
FORCE    200     90      0       125.    -1.     0.      0.
FORCE    300     55      0       10.     0.     -1.      0.
FORCE    300     90      0       10.     0.     -1.      0.
FORCE    400     85      0       20.     0.     -1.      0.
FORCE    400     86      0       20.     0.     -1.      0.
FORCE    400     87      0       20.     0.     -1.      0.
FORCE    400     88      0       20.     0.     -1.      0.
FORCE    400     89      0       20.     0.     -1.      0.
$
PLOAD2       500-125.0         8
PLOAD2       500-125.0         9
```

```
PLOAD2       500-125.0       18
PLOAD2       500-125.0       19
PLOAD2       500-125.0       28
PLOAD2       500-125.0       29
PLOAD2       500-125.0       38
PLOAD2       500-125.0       39
$
ENDDATA
```

10.4 ATS4m4.bdf

```
$ Linear Static Analysis
SOL 101
TIME 600
CEND
$
SEALL = ALL
SUPER = ALL
TITLE = Nastran job EAS test case ATS4m4
ECHO = NONE
MAXLINES = 999999999
GPFORCE(PUNCH) = ALL
$
SUBCASE 1
   SUBTITLE=subcase1
   SPC = 11
   LOAD = 21
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
SUBCASE 2
   SUBTITLE=subcase2
   SPC = 12
   LOAD = 22
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
SUBCASE 3
   SUBTITLE=subcase3
   SPC = 12
   LOAD = 23
   DISPLACEMENT(PUNCH,SORT1,REAL)=ALL
   SPCFORCES(PUNCH,SORT1,REAL)=ALL
   STRESS(PUNCH,SORT1,REAL,VONMISES,BILIN)=ALL
$
BEGIN BULK
PARAM   POST    -1
PARAM   AUTOSPC YES
PARAM   PRTMAXIM YES
$
PSOLID  1       1       0
$
CHEXA   1       1       1       2       7       6       16      17
        22      21
CHEXA   2       1       2       3       8       7       17      18
        23      22
CHEXA   3       1       3       4       9       8       18      19
        24      23
CHEXA   4       1       4       5       10      9       19      20
        25      24
CHEXA   5       1       6       7       12      11      21      22
        27      26
CHEXA   6       1       7       8       13      12      22      23
        28      27
CHEXA   7       1       8       9       14      13      23      24
        29      28
CHEXA   8       1       9       10      15      14      24      25
        30      29
CHEXA   9       1       16      17      22      21      31      32
        37      36
CHEXA   10      1       17      18      23      22      32      33
        38      37
CHEXA   11      1       18      19      24      23      33      34
        39      38
CHEXA   12      1       19      20      25      24      34      35
        40      39
CHEXA   13      1       21      22      27      26      36      37
        42      41
CHEXA   14      1       22      23      28      27      37      38
        43      42
```

```
CHEXA   15      1       23      24      29      28      38      39
        44      43
CHEXA   16      1       24      25      30      29      39      40
        45      44
CHEXA   17      1       31      32      37      36      46      47
        52      51
CHEXA   18      1       32      33      38      37      47      48
        53      52
CHEXA   19      1       33      34      39      38      48      49
        54      53
CHEXA   20      1       34      35      40      39      49      50
        55      54
CHEXA   21      1       36      37      42      41      51      52
        57      56
CHEXA   22      1       37      38      43      42      52      53
        58      57
CHEXA   23      1       38      39      44      43      53      54
        59      58
CHEXA   24      1       39      40      45      44      54      55
        60      59
CHEXA   25      1       46      47      52      51      61      62
        67      66
CHEXA   26      1       47      48      53      52      62      63
        68      67
CHEXA   27      1       48      49      54      53      63      64
        69      68
CHEXA   28      1       49      50      55      54      64      65
        70      69
CHEXA   29      1       51      52      57      56      66      67
        72      71
CHEXA   30      1       52      53      58      57      67      68
        73      72
CHEXA   31      1       53      54      59      58      68      69
        74      73
CHEXA   32      1       54      55      60      59      69      70
        75      74
$
CTETRA  33      1       76      83      111     77
CTETRA  34      1       118     111     83      119
CTETRA  35      1       84      77      119     83
CTETRA  36      1       112     119     77      111
CTETRA  37      1       77      83      111     119
CTETRA  38      1       84      119     77      85
CTETRA  39      1       112     77      119     113
CTETRA  40      1       78      113     85      77
CTETRA  41      1       120     85      113     119
CTETRA  42      1       77      85      119     113
CTETRA  43      1       78      85      113     79
CTETRA  44      1       120     113     85      121
CTETRA  45      1       86      79      121     85
CTETRA  46      1       114     121     79      113
CTETRA  47      1       79      85      113     121
CTETRA  48      1       86      121     79      87
CTETRA  49      1       114     79      121     115
CTETRA  50      1       80      115     87      79
CTETRA  51      1       122     87      115     121
CTETRA  52      1       79      87      121     115
CTETRA  53      1       80      87      115     81
CTETRA  54      1       122     115     87      123
CTETRA  55      1       88      81      123     87
CTETRA  56      1       116     123     81      115
CTETRA  57      1       81      87      115     123
CTETRA  58      1       88      123     81      64
CTETRA  59      1       116     81      123     70
CTETRA  60      1       65      70      64      81
CTETRA  61      1       69      64      70      123
CTETRA  62      1       81      64      123     70
CTETRA  63      1       90      125     83      91
CTETRA  64      1       118     83      125     119
CTETRA  65      1       84      119     91      83
CTETRA  66      1       126     91      119     125
```

CTETRA	67	1	83	91	125	119
CTETRA	68	1	84	91	119	85
CTETRA	69	1	126	119	91	127
CTETRA	70	1	92	85	127	91
CTETRA	71	1	120	127	85	119
CTETRA	72	1	85	91	119	127
CTETRA	73	1	92	127	85	93
CTETRA	74	1	120	85	127	121
CTETRA	75	1	86	121	93	85
CTETRA	76	1	128	93	121	127
CTETRA	77	1	85	93	127	121
CTETRA	78	1	86	93	121	87
CTETRA	79	1	128	121	93	129
CTETRA	80	1	94	87	129	93
CTETRA	81	1	122	129	87	121
CTETRA	82	1	87	93	121	129
CTETRA	83	1	94	129	87	95
CTETRA	84	1	122	87	129	123
CTETRA	85	1	88	123	95	87
CTETRA	86	1	130	95	123	129
CTETRA	87	1	87	95	129	123
CTETRA	88	1	88	95	123	64
CTETRA	89	1	130	123	95	68
CTETRA	90	1	63	64	68	95
CTETRA	91	1	69	68	64	123
CTETRA	92	1	64	95	123	68
CTETRA	93	1	90	97	125	91
CTETRA	94	1	132	125	97	133
CTETRA	95	1	98	91	133	97
CTETRA	96	1	126	133	91	125
CTETRA	97	1	91	97	125	133
CTETRA	98	1	98	133	91	99
CTETRA	99	1	126	91	133	127
CTETRA	100	1	92	127	99	91
CTETRA	101	1	134	99	127	133
CTETRA	102	1	91	99	133	127
CTETRA	103	1	92	99	127	93
CTETRA	104	1	134	127	99	135
CTETRA	105	1	100	93	135	99
CTETRA	106	1	128	135	93	127
CTETRA	107	1	93	99	127	135
CTETRA	108	1	100	135	93	101
CTETRA	109	1	128	93	135	129
CTETRA	110	1	94	129	101	93
CTETRA	111	1	136	101	129	135
CTETRA	112	1	93	101	135	129
CTETRA	113	1	94	101	129	95
CTETRA	114	1	136	129	101	137
CTETRA	115	1	102	95	137	101
CTETRA	116	1	130	137	95	129
CTETRA	117	1	95	101	129	137
CTETRA	118	1	102	137	95	62
CTETRA	119	1	130	95	137	68
CTETRA	120	1	63	68	62	95
CTETRA	121	1	67	62	68	137
CTETRA	122	1	95	62	137	68
CTETRA	123	1	104	139	97	105
CTETRA	124	1	132	97	139	133
CTETRA	125	1	98	133	105	97
CTETRA	126	1	140	105	133	139
CTETRA	127	1	97	105	139	133
CTETRA	128	1	98	105	133	99
CTETRA	129	1	140	133	105	141
CTETRA	130	1	106	99	141	105
CTETRA	131	1	134	141	99	133
CTETRA	132	1	99	105	133	141
CTETRA	133	1	106	141	99	107
CTETRA	134	1	134	99	141	135
CTETRA	135	1	100	135	107	99
CTETRA	136	1	142	107	135	141
CTETRA	137	1	99	107	141	135

CTETRA	138	1	100	107	135	101
CTETRA	139	1	142	135	107	143
CTETRA	140	1	108	101	143	107
CTETRA	141	1	136	143	101	135
CTETRA	142	1	101	107	135	143
CTETRA	143	1	108	143	101	109
CTETRA	144	1	136	101	143	137
CTETRA	145	1	102	137	109	101
CTETRA	146	1	144	109	137	143
CTETRA	147	1	101	109	143	137
CTETRA	148	1	102	109	137	62
CTETRA	149	1	144	137	109	66
CTETRA	150	1	61	62	66	109
CTETRA	151	1	67	66	62	137
CTETRA	152	1	62	109	137	66
CTETRA	153	1	118	153	111	119
CTETRA	154	1	146	111	153	147
CTETRA	155	1	112	147	119	111
CTETRA	156	1	154	119	147	153
CTETRA	157	1	111	119	153	147
CTETRA	158	1	112	119	147	113
CTETRA	159	1	154	147	119	155
CTETRA	160	1	120	113	155	119
CTETRA	161	1	148	155	113	147
CTETRA	162	1	113	119	147	155
CTETRA	163	1	120	155	113	121
CTETRA	164	1	148	113	155	149
CTETRA	165	1	114	149	121	113
CTETRA	166	1	156	121	149	155
CTETRA	167	1	113	121	155	149
CTETRA	168	1	114	121	149	115
CTETRA	169	1	156	149	121	157
CTETRA	170	1	122	115	157	121
CTETRA	171	1	150	157	115	149
CTETRA	172	1	115	121	149	157
CTETRA	173	1	122	157	115	123
CTETRA	174	1	150	115	157	151
CTETRA	175	1	116	151	123	115
CTETRA	176	1	158	123	151	157
CTETRA	177	1	115	123	157	151
CTETRA	178	1	116	123	151	70
CTETRA	179	1	158	151	123	74
CTETRA	180	1	69	70	74	123
CTETRA	181	1	75	74	70	151
CTETRA	182	1	70	123	151	74
CTETRA	183	1	118	125	153	119
CTETRA	184	1	160	153	125	161
CTETRA	185	1	126	119	161	125
CTETRA	186	1	154	161	119	153
CTETRA	187	1	119	125	153	161
CTETRA	188	1	126	161	119	127
CTETRA	189	1	154	119	161	155
CTETRA	190	1	120	155	127	119
CTETRA	191	1	162	127	155	161
CTETRA	192	1	119	127	161	155
CTETRA	193	1	120	127	155	121
CTETRA	194	1	162	155	127	163
CTETRA	195	1	128	121	163	127
CTETRA	196	1	156	163	121	155
CTETRA	197	1	121	127	155	163
CTETRA	198	1	128	163	121	129
CTETRA	199	1	156	121	163	157
CTETRA	200	1	122	157	129	121
CTETRA	201	1	164	129	157	163
CTETRA	202	1	121	129	163	157
CTETRA	203	1	122	129	157	123
CTETRA	204	1	164	157	129	165
CTETRA	205	1	130	123	165	129
CTETRA	206	1	158	165	123	157
CTETRA	207	1	123	129	157	165
CTETRA	208	1	130	165	123	68

```
CTETRA  209  1  158  123  165  74
CTETRA  210  1  69   74   68   123
CTETRA  211  1  73   68   74   165
CTETRA  212  1  123  68   165  74
CTETRA  213  1  132  167  125  133
CTETRA  214  1  160  125  167  161
CTETRA  215  1  126  161  133  125
CTETRA  216  1  168  133  161  167
CTETRA  217  1  125  133  167  161
CTETRA  218  1  126  133  161  127
CTETRA  219  1  168  161  133  169
CTETRA  220  1  134  127  169  133
CTETRA  221  1  162  169  127  161
CTETRA  222  1  127  133  161  169
CTETRA  223  1  134  169  127  135
CTETRA  224  1  162  127  169  163
CTETRA  225  1  128  163  135  127
CTETRA  226  1  170  135  163  169
CTETRA  227  1  127  135  169  163
CTETRA  228  1  128  135  163  129
CTETRA  229  1  170  163  135  171
CTETRA  230  1  136  129  171  135
CTETRA  231  1  164  171  129  163
CTETRA  232  1  129  135  163  171
CTETRA  233  1  136  171  129  137
CTETRA  234  1  164  129  171  165
CTETRA  235  1  130  165  137  129
CTETRA  236  1  172  137  165  171
CTETRA  237  1  129  137  171  165
CTETRA  238  1  130  137  165  68
CTETRA  239  1  172  165  137  72
CTETRA  240  1  67   68   72   137
CTETRA  241  1  73   72   68   165
CTETRA  242  1  68   137  165  72
CTETRA  243  1  132  139  167  133
CTETRA  244  1  174  167  139  175
CTETRA  245  1  140  133  175  139
CTETRA  246  1  168  175  133  167
CTETRA  247  1  133  139  167  175
CTETRA  248  1  140  175  133  141
CTETRA  249  1  168  133  175  169
CTETRA  250  1  134  169  141  133
CTETRA  251  1  176  141  169  175
CTETRA  252  1  133  141  175  169
CTETRA  253  1  134  141  169  135
CTETRA  254  1  176  169  141  177
CTETRA  255  1  142  135  177  141
CTETRA  256  1  170  177  135  169
CTETRA  257  1  135  141  169  177
CTETRA  258  1  142  177  135  143
CTETRA  259  1  170  135  177  171
CTETRA  260  1  136  171  143  135
CTETRA  261  1  178  143  171  177
CTETRA  262  1  135  143  177  171
CTETRA  263  1  136  143  171  137
CTETRA  264  1  178  171  143  179
CTETRA  265  1  144  137  179  143
CTETRA  266  1  172  179  137  171
CTETRA  267  1  137  143  171  179
CTETRA  268  1  144  179  137  66
CTETRA  269  1  172  137  179  72
CTETRA  270  1  67   72   66   137
CTETRA  271  1  71   66   72   179
CTETRA  272  1  137  66   179  72
$
CPENTA  273  1  189  188  181  224  223  216
CPENTA  274  1  181  182  189  216  217  224
CPENTA  275  1  182  183  189  217  218  224
CPENTA  276  1  190  189  183  225  224  218
CPENTA  277  1  191  190  183  226  225  218
CPENTA  278  1  183  184  191  218  219  226
```

CPENTA	279	1	184	185	191	219	220	226
CPENTA	280	1	192	191	185	227	226	220
CPENTA	281	1	193	192	185	228	227	220
CPENTA	282	1	185	186	193	220	221	228
CPENTA	283	1	186	76	193	221	111	228
CPENTA	284	1	83	193	76	118	228	111
CPENTA	285	1	188	189	195	223	224	230
CPENTA	286	1	196	195	189	231	230	224
CPENTA	287	1	197	196	189	232	231	224
CPENTA	288	1	189	190	197	224	225	232
CPENTA	289	1	190	191	197	225	226	232
CPENTA	290	1	198	197	191	233	232	226
CPENTA	291	1	199	198	191	234	233	226
CPENTA	292	1	191	192	199	226	227	234
CPENTA	293	1	192	193	199	227	228	234
CPENTA	294	1	200	199	193	235	234	228
CPENTA	295	1	90	200	193	125	235	228
CPENTA	296	1	193	83	90	228	118	125
CPENTA	297	1	203	202	195	238	237	230
CPENTA	298	1	195	196	203	230	231	238
CPENTA	299	1	196	197	203	231	232	238
CPENTA	300	1	204	203	197	239	238	232
CPENTA	301	1	205	204	197	240	239	232
CPENTA	302	1	197	198	205	232	233	240
CPENTA	303	1	198	199	205	233	234	240
CPENTA	304	1	206	205	199	241	240	234
CPENTA	305	1	207	206	199	242	241	234
CPENTA	306	1	199	200	207	234	235	242
CPENTA	307	1	200	90	207	235	125	242
CPENTA	308	1	97	207	90	132	242	125
CPENTA	309	1	202	203	209	237	238	244
CPENTA	310	1	210	209	203	245	244	238
CPENTA	311	1	211	210	203	246	245	238
CPENTA	312	1	203	204	211	238	239	246
CPENTA	313	1	204	205	211	239	240	246
CPENTA	314	1	212	211	205	247	246	240
CPENTA	315	1	213	212	205	248	247	240
CPENTA	316	1	205	206	213	240	241	248
CPENTA	317	1	206	207	213	241	242	248
CPENTA	318	1	214	213	207	249	248	242
CPENTA	319	1	104	214	207	139	249	242
CPENTA	320	1	207	97	104	242	132	139
CPENTA	321	1	224	223	216	259	258	251
CPENTA	322	1	216	217	224	251	252	259
CPENTA	323	1	217	218	224	252	253	259
CPENTA	324	1	225	224	218	260	259	253
CPENTA	325	1	226	225	218	261	260	253
CPENTA	326	1	218	219	226	253	254	261
CPENTA	327	1	219	220	226	254	255	261
CPENTA	328	1	227	226	220	262	261	255
CPENTA	329	1	228	227	220	263	262	255
CPENTA	330	1	220	221	228	255	256	263
CPENTA	331	1	221	111	228	256	146	263
CPENTA	332	1	118	228	111	153	263	146
CPENTA	333	1	223	224	230	258	259	265
CPENTA	334	1	231	230	224	266	265	259
CPENTA	335	1	232	231	224	267	266	259
CPENTA	336	1	224	225	232	259	260	267
CPENTA	337	1	225	226	232	260	261	267
CPENTA	338	1	233	232	226	268	267	261
CPENTA	339	1	234	233	226	269	268	261
CPENTA	340	1	226	227	234	261	262	269
CPENTA	341	1	227	228	234	262	263	269
CPENTA	342	1	235	234	228	270	269	263
CPENTA	343	1	125	235	228	160	270	263
CPENTA	344	1	228	118	125	263	153	160
CPENTA	345	1	238	237	230	273	272	265
CPENTA	346	1	230	231	238	265	266	273
CPENTA	347	1	231	232	238	266	267	273
CPENTA	348	1	239	238	232	274	273	267
CPENTA	349	1	240	239	232	275	274	267

```
CPENTA   350     1       232     233     240     267     268     275
CPENTA   351     1       233     234     240     268     269     275
CPENTA   352     1       241     240     234     276     275     269
CPENTA   353     1       242     241     234     277     276     269
CPENTA   354     1       234     235     242     269     270     277
CPENTA   355     1       235     125     242     270     160     277
CPENTA   356     1       132     242     125     167     277     160
CPENTA   357     1       237     238     244     272     273     279
CPENTA   358     1       245     244     238     280     279     273
CPENTA   359     1       246     245     238     281     280     273
CPENTA   360     1       238     239     246     273     274     281
CPENTA   361     1       239     240     246     274     275     281
CPENTA   362     1       247     246     240     282     281     275
CPENTA   363     1       248     247     240     283     282     275
CPENTA   364     1       240     241     248     275     276     283
CPENTA   365     1       241     242     248     276     277     283
CPENTA   366     1       249     248     242     284     283     277
CPENTA   367     1       139     249     242     174     284     277
CPENTA   368     1       242     132     139     277     167     174
$
MAT1     1       1.+7            .33     2.54-4  1.3-5   70.
$
GRID*    1                               0.              -4.
*       -3.43152-8
GRID*    2                               0.              -3.
*       -3.43152-8
GRID*    3                               0.              -2.
*       -3.43152-8
GRID*    4                               0.              -1.
*       -3.43152-8
GRID*    5                               0.              -6.01376-9
*       -3.43152-8
GRID     6               0.      -4.     1.
GRID     7               0.      -3.     1.
GRID     8               0.      -2.     1.
GRID     9               0.      -1.     1.
GRID*    10                              0.              -6.01376-9
*        1.
GRID     11              0.      -4.     2.
GRID     12              0.      -3.     2.
GRID     13              0.      -2.     2.
GRID     14              0.      -1.     2.
GRID*    15                              0.              -6.01376-9
*        2.
GRID*    16                              1.              -4.
*       -2.53545-8
GRID*    17                              1.              -3.
*       -2.09009-8
GRID*    18                              1.              -2.
*       -3.74229-8
GRID*    19                              1.              -1.
*       -5.3945-8
GRID*    20                              1.              -2.86429-9
*       -4.94913-8
GRID     21              1.      -4.     1.
GRID     22              1.      -3.     1.
GRID     23              1.      -2.     1.
GRID     24              1.      -1.     1.
GRID*    25                              1.              -3.28417-9
*        1.
GRID     26              1.      -4.     2.
GRID     27              1.      -3.     2.
GRID     28              1.      -2.     2.
GRID     29              1.      -1.     2.
GRID*    30                              1.              -3.70404-9
*        2.
GRID*    31                              2.              -4.
*       -1.63938-8
GRID*    32                              2.              -3.
*       -7.48651-9
GRID*    33                              2.              -2.
```

```
*       -4.05306-8
GRID*    34                         2.             -1.
*       -7.35747-8
GRID*    35                         2.              2.85177-10
*       -6.46674-8
GRID     36             2.      -4.     1.
GRID     37             2.      -3.     1.
GRID     38             2.      -2.     1.
GRID     39             2.      -1.     1.
GRID*    40                         2.             -5.54573-10
*        1.
GRID     41             2.      -4.     2.
GRID     42             2.      -3.     2.
GRID     43             2.      -2.     2.
GRID     44             2.      -1.     2.
GRID*    45                         2.             -1.39432-9
*        2.
GRID*    46                         3.             -4.
*       -7.43315-9
GRID*    47                         3.             -3.
*        5.92784-9
GRID*    48                         3.             -2.
*       -4.36383-8
GRID*    49                         3.             -1.
*       -9.32045-8
GRID*    50                         3.              3.43465-9
*       -7.98435-8
GRID     51             3.      -4.     1.
GRID     52             3.      -3.     1.
GRID     53             3.      -2.     1.
GRID     54             3.      -1.     1.
GRID*    55                         3.              2.17502-9
*        1.
GRID     56             3.      -4.     2.
GRID     57             3.      -3.     2.
GRID     58             3.      -2.     2.
GRID     59             3.      -1.     2.
GRID*    60                         3.              9.154-10
*        2.
GRID*    61                         4.             -4.
*        1.52753-9
GRID*    62                         4.             -3.
*        1.93422-8
GRID*    63                         4.             -2.
*       -4.6746-8
GRID*    64                         4.             -1.
*       -1.12834-7
GRID*    65                         4.              6.58412-9
*       -9.50196-8
GRID     66             4.      -4.     1.
GRID     67             4.      -3.     1.
GRID     68             4.      -2.     1.
GRID     69             4.      -1.     1.
GRID*    70                         4.              4.90462-9
*        1.
GRID     71             4.      -4.     2.
GRID     72             4.      -3.     2.
GRID     73             4.      -2.     2.
GRID     74             4.      -1.     2.
GRID*    75                         4.              3.22512-9
*        2.
GRID*    76                         10.            -5.27577-8
*        0.
GRID*    77                         9.             -5.27577-8
*        0.
GRID*    78                         8.             -5.27577-8
*        0.
GRID*    79                         7.             -5.27577-8
*        0.
GRID*    80                         6.             -5.27577-8
*        0.
```

```
GRID*    81                      5.              -5.27577-8
*        0.
GRID     83              10.    -1.     0.
GRID     84              9.     -1.     0.
GRID     85              8.     -1.     0.
GRID     86              7.     -1.     0.
GRID     87              6.     -1.     0.
GRID     88              5.     -1.     0.
GRID     90              10.    -2.     0.
GRID     91              9.     -2.     0.
GRID     92              8.     -2.     0.
GRID     93              7.     -2.     0.
GRID     94              6.     -2.     0.
GRID     95              5.     -2.     0.
GRID     97              10.    -3.     0.
GRID     98              9.     -3.     0.
GRID     99              8.     -3.     0.
GRID     100             7.     -3.     0.
GRID     101             6.     -3.     0.
GRID     102             5.     -3.     0.
GRID     104             10.    -4.     0.
GRID     105             9.     -4.     0.
GRID     106             8.     -4.     0.
GRID     107             7.     -4.     0.
GRID     108             6.     -4.     0.
GRID     109             5.     -4.     0.
GRID*    111                     10.             -5.27577-8
*        1.
GRID*    112                     9.              -5.27577-8
*        1.
GRID*    113                     8.              -5.27577-8
*        1.
GRID*    114                     7.              -5.27577-8
*        1.
GRID*    115                     6.              -5.27577-8
*        1.
GRID*    116                     5.              -5.27577-8
*        1.
GRID     118             10.    -1.     1.
GRID     119             9.     -1.     1.
GRID     120             8.     -1.     1.
GRID     121             7.     -1.     1.
GRID     122             6.     -1.     1.
GRID     123             5.     -1.     1.
GRID     125             10.    -2.     1.
GRID     126             9.     -2.     1.
GRID     127             8.     -2.     1.
GRID     128             7.     -2.     1.
GRID     129             6.     -2.     1.
GRID     130             5.     -2.     1.
GRID     132             10.    -3.     1.
GRID     133             9.     -3.     1.
GRID     134             8.     -3.     1.
GRID     135             7.     -3.     1.
GRID     136             6.     -3.     1.
GRID     137             5.     -3.     1.
GRID     139             10.    -4.     1.
GRID     140             9.     -4.     1.
GRID     141             8.     -4.     1.
GRID     142             7.     -4.     1.
GRID     143             6.     -4.     1.
GRID     144             5.     -4.     1.
GRID*    146                     10.             -5.27577-8
*        2.
GRID*    147                     9.              -5.27577-8
*        2.
GRID*    148                     8.              -5.27577-8
*        2.
GRID*    149                     7.              -5.27577-8
*        2.
GRID*    150                     6.              -5.27577-8
```

```
*       2.
GRID*   151                             5.              -5.27577-8
*       2.
GRID    153             10.     -1.     2.
GRID    154             9.      -1.     2.
GRID    155             8.      -1.     2.
GRID    156             7.      -1.     2.
GRID    157             6.      -1.     2.
GRID    158             5.      -1.     2.
GRID    160             10.     -2.     2.
GRID    161             9.      -2.     2.
GRID    162             8.      -2.     2.
GRID    163             7.      -2.     2.
GRID    164             6.      -2.     2.
GRID    165             5.      -2.     2.
GRID    167             10.     -3.     2.
GRID    168             9.      -3.     2.
GRID    169             8.      -3.     2.
GRID    170             7.      -3.     2.
GRID    171             6.      -3.     2.
GRID    172             5.      -3.     2.
GRID    174             10.     -4.     2.
GRID    175             9.      -4.     2.
GRID    176             8.      -4.     2.
GRID    177             7.      -4.     2.
GRID    178             6.      -4.     2.
GRID    179             5.      -4.     2.
GRID*   181                             16.             -5.27577-8
*       0.
GRID*   182                             15.             -5.27577-8
*       0.
GRID*   183                             14.             -5.27577-8
*       0.
GRID*   184                             13.             -5.27577-8
*       0.
GRID*   185                             12.             -5.27577-8
*       0.
GRID*   186                             11.             -5.27577-8
*       0.
GRID    188             16.     -1.     0.
GRID    189             15.     -1.     0.
GRID    190             14.     -1.     0.
GRID    191             13.     -1.     0.
GRID    192             12.     -1.     0.
GRID    193             11.     -1.     0.
GRID    195             16.     -2.     0.
GRID    196             15.     -2.     0.
GRID    197             14.     -2.     0.
GRID    198             13.     -2.     0.
GRID    199             12.     -2.     0.
GRID    200             11.     -2.     0.
GRID    202             16.     -3.     0.
GRID    203             15.     -3.     0.
GRID    204             14.     -3.     0.
GRID    205             13.     -3.     0.
GRID    206             12.     -3.     0.
GRID    207             11.     -3.     0.
GRID    209             16.     -4.     0.
GRID    210             15.     -4.     0.
GRID    211             14.     -4.     0.
GRID    212             13.     -4.     0.
GRID    213             12.     -4.     0.
GRID    214             11.     -4.     0.
GRID*   216                             16.             -5.27577-8
*       1.
GRID*   217                             15.             -5.27577-8
*       1.
GRID*   218                             14.             -5.27577-8
*       1.
GRID*   219                             13.             -5.27577-8
*       1.
```

```
GRID*   220                     12.             -5.27577-8
*       1.
GRID*   221                     11.             -5.27577-8
*       1.
GRID    223             16.     -1.     1.
GRID    224             15.     -1.     1.
GRID    225             14.     -1.     1.
GRID    226             13.     -1.     1.
GRID    227             12.     -1.     1.
GRID    228             11.     -1.     1.
GRID    230             16.     -2.     1.
GRID    231             15.     -2.     1.
GRID    232             14.     -2.     1.
GRID    233             13.     -2.     1.
GRID    234             12.     -2.     1.
GRID    235             11.     -2.     1.
GRID    237             16.     -3.     1.
GRID    238             15.     -3.     1.
GRID    239             14.     -3.     1.
GRID    240             13.     -3.     1.
GRID    241             12.     -3.     1.
GRID    242             11.     -3.     1.
GRID    244             16.     -4.     1.
GRID    245             15.     -4.     1.
GRID    246             14.     -4.     1.
GRID    247             13.     -4.     1.
GRID    248             12.     -4.     1.
GRID    249             11.     -4.     1.
GRID*   251                     16.             -5.27577-8
*       2.
GRID*   252                     15.             -5.27577-8
*       2.
GRID*   253                     14.             -5.27577-8
*       2.
GRID*   254                     13.             -5.27577-8
*       2.
GRID*   255                     12.             -5.27577-8
*       2.
GRID*   256                     11.             -5.27577-8
*       2.
GRID    258             16.     -1.     2.
GRID    259             15.     -1.     2.
GRID    260             14.     -1.     2.
GRID    261             13.     -1.     2.
GRID    262             12.     -1.     2.
GRID    263             11.     -1.     2.
GRID    265             16.     -2.     2.
GRID    266             15.     -2.     2.
GRID    267             14.     -2.     2.
GRID    268             13.     -2.     2.
GRID    269             12.     -2.     2.
GRID    270             11.     -2.     2.
GRID    272             16.     -3.     2.
GRID    273             15.     -3.     2.
GRID    274             14.     -3.     2.
GRID    275             13.     -3.     2.
GRID    276             12.     -3.     2.
GRID    277             11.     -3.     2.
GRID    279             16.     -4.     2.
GRID    280             15.     -4.     2.
GRID    281             14.     -4.     2.
GRID    282             13.     -4.     2.
GRID    283             12.     -4.     2.
GRID    284             11.     -4.     2.
$
SPCADD  11      100     101
SPCADD  12      100     102
$
LOAD    21      1.      1.      200
LOAD    22      1.      1.      300     1.      400
LOAD    23      1.      1.      200     1.      300     1.      400
```

```
$
SPC1     100     123     8
SPC1     100     3       6
SPC1     101     1       1       THRU    7
SPC1     101     1       9       THRU    15
SPC1     102     123     1       THRU    7
SPC1     102     123     9       THRU    15
$
FORCE    200     181     0       31.25   -1.     0.      0.
FORCE    200     209     0       31.25   -1.     0.      0.
FORCE    200     251     0       31.25   -1.     0.      0.
FORCE    200     279     0       31.25   -1.     0.      0.
FORCE    200     188     0       62.50   -1.     0.      0.
FORCE    200     195     0       62.50   -1.     0.      0.
FORCE    200     202     0       62.50   -1.     0.      0.
FORCE    200     216     0       62.50   -1.     0.      0.
FORCE    200     244     0       62.50   -1.     0.      0.
FORCE    200     258     0       62.50   -1.     0.      0.
FORCE    200     265     0       62.50   -1.     0.      0.
FORCE    200     272     0       62.50   -1.     0.      0.
FORCE    200     223     0       125.0   -1.     0.      0.
FORCE    200     230     0       125.0   -1.     0.      0.
FORCE    200     237     0       125.0   -1.     0.      0.
FORCE        300      76       01.0     0.0     -2.5    0.0
FORCE        300     146       01.0     0.0     -2.5    0.0
FORCE        300     181       01.0     0.0     -2.5    0.0
FORCE        300     251       01.0     0.0     -2.5    0.0
FORCE        300     111       01.0     0.0     -5.0    0.0
FORCE        300     186       01.0     0.0     -5.0    0.0
FORCE        300     183       01.0     0.0     -5.0    0.0
FORCE        300     184       01.0     0.0     -5.0    0.0
FORCE        300     185       01.0     0.0     -5.0    0.0
FORCE        300     182       01.0     0.0     -5.0    0.0
FORCE        300     216       01.0     0.0     -5.0    0.0
FORCE        300     252       01.0     0.0     -5.0    0.0
FORCE        300     253       01.0     0.0     -5.0    0.0
FORCE        300     254       01.0     0.0     -5.0    0.0
FORCE        300     255       01.0     0.0     -5.0    0.0
FORCE        300     256       01.0     0.0     -5.0    0.0
FORCE        400     217       01.0     0.0     -10.0   0.0
FORCE        400     218       01.0     0.0     -10.0   0.0
FORCE        400     219       01.0     0.0     -10.0   0.0
FORCE        400     220       01.0     0.0     -10.0   0.0
FORCE        400     221       01.0     0.0     -10.0   0.0
$
ENDDATA
```

11 Appendix B – STEP Part21 Output File Full Listing

The following pages list the full STEP Part21 file contents for the 4 pilot models. This section will be updated as the STEP converters mature and required corrections are incorporated into this document. As such, the actual instance identifiers in these files may not match the ones used in the figures and tables within the body of this document.

11.1 ATS1m4.stp

```
ISO-10303-21;
HEADER;

/****************************************************************************************
* Creation date:                   Tue Feb 07 13:53:28 2017
* Schema:                          AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF
****************************************************************************************/

FILE_DESCRIPTION((''),'2;1');
FILE_NAME(
'O:\\projects\\Crystal\\software_FEM\\converters\\examples\\db\\e.DataRepository.Fem_Model'

,
          '2017-02-07T13:53:28',
          ('ANONYMOUS USER'),
          ('ANONYMOUS ORGANISATION'),
          'EDMsix Version 2.0100.a14 Jan 31 2017',
          '',
          '');
FILE_SCHEMA(('AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF'));
ENDSEC;

DATA;
#1= APPLICATION_CONTEXT(
'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF');
#3= ID_ATTRIBUTE('to be decided',#1);
#4= DESCRIPTION_ATTRIBUTE('to be decided',#1);
#5= APPLICATION_PROTOCOL_DEFINITION('international standard',
'ap209_multidisciplinary_analysis_and_design',2011,#1);
#6= DIMENSIONAL_EXPONENTS(0.,0.,0.,0.,0.,0.,0.);
#7= /* LENGTH_UNIT+SI_UNIT */(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT($,
.METRE.));
#8= /* MASS_UNIT+SI_UNIT */(MASS_UNIT()NAMED_UNIT(*)SI_UNIT(.KILO.,
.GRAM.));
#9= /* SI_UNIT+TIME_UNIT */(NAMED_UNIT(*)SI_UNIT($,.SECOND.)TIME_UNIT()
);
#10= /* SI_UNIT+THERMODYNAMIC_TEMPERATURE_UNIT */(NAMED_UNIT(*)SI_UNIT($
,.DEGREE_CELSIUS.)THERMODYNAMIC_TEMPERATURE_UNIT());
#11= /* PLANE_ANGLE_UNIT+SI_UNIT */(NAMED_UNIT(*)PLANE_ANGLE_UNIT()
SI_UNIT($,.RADIAN.));
#12= SI_FORCE_UNIT((#13,#14,#15),*,$,.NEWTON.);
#13= DERIVED_UNIT_ELEMENT(#8,1.);
#14= DERIVED_UNIT_ELEMENT(#7,1.);
#15= DERIVED_UNIT_ELEMENT(#9,-2.);
#17= SI_FREQUENCY_UNIT((#18),*,$,.HERTZ.);
#18= DERIVED_UNIT_ELEMENT(#9,-1.);
#20= SI_PRESSURE_UNIT((#13,#21,#15),*,$,.PASCAL.);
#21= DERIVED_UNIT_ELEMENT(#7,-1.);
#23= SI_ENERGY_UNIT((#13,#24,#15),*,$,.JOULE.);
#24= DERIVED_UNIT_ELEMENT(#7,2.);
#26= SI_POWER_UNIT((#13,#24,#27),*,$,.WATT.);
#27= DERIVED_UNIT_ELEMENT(#9,-3.);
#29= DIRECTION('RefX',(1.,0.,0.));
#31= DIRECTION('AxisZ',(0.,0.,1.));
#33= CARTESIAN_POINT('',(0.,0.,0.));
#35= FEA_AXIS2_PLACEMENT_3D('0',#33,#31,#29,.CARTESIAN.,
'FEA_BASIC_COORD_SYSTEM.0');
#36= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT(
'FEA_BASIC_COORD_SYSTEM.0','3d'));
#39= DIRECTION('RefX',(1.,0.,0.));
#41= DIRECTION('AxisZ',(0.,0.,1.));
#43= CARTESIAN_POINT('',(0.,0.,0.));
#45= FEA_AXIS2_PLACEMENT_3D('1',#43,#41,#39,.CARTESIAN.,'CORD2R.1');
#46= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT('CORD2R.1','3d'
));
```

```
#49= ORGANIZATION($,'default-organization',$);
#50= PRODUCT('default-id.0','default-fea-part','',(#52));
#52= PRODUCT_CONTEXT('design_context',#1,'design_context');
#54= PRODUCT_RELATED_PRODUCT_CATEGORY('part',$,(#50));
#55= ID_ATTRIBUTE('',#54);
#57= APPLIED_IDENTIFICATION_ASSIGNMENT('default-id.0',#58,(#60,#66,#50)
);
#58= IDENTIFICATION_ROLE('default-role',$);
#59= PRODUCT_RELATED_PRODUCT_CATEGORY('product',$,(#50));
#60= PRODUCT_DEFINITION_FORMATION('default-id.0','',#50);
#61= GROUP('id',$);
#62= APPLIED_CLASSIFICATION_ASSIGNMENT(#61,#64,(#57));
#64= CLASSIFICATION_ROLE('',$);
#66= PRODUCT_DEFINITION('default-id.0',$,#60,#67);
#67= PRODUCT_DEFINITION_CONTEXT('assembly definition',#1,
'life cycle stage');
#68= APPLIED_CLASSIFICATION_ASSIGNMENT(#69,#70,(#66));
#69= GROUP('default-gpvv-type',$);
#70= CLASSIFICATION_ROLE('view_type',$);
#73= ORGANIZATION_ROLE('id context');
#74= APPLIED_ORGANIZATION_ASSIGNMENT(#49,#73,(#57));
#76= NAME_ATTRIBUTE('',#66);
#77= FEA_MODEL_3D('Identification',(#35),#36,
'NASTRAN BDF Converter v0.0.0',('NASTRAN'),'AnalysisModelType');
#80= STRUCTURAL_RESPONSE_PROPERTY_DEFINITION_REPRESENTATION(#81,#77);
#81= STRUCTURAL_RESPONSE_PROPERTY('structural_response_property_id_1',$,
#82);
#82= FEA_MODEL_DEFINITION('fea_model_definition_id_1',$,#83,.U.);
#83= PRODUCT_DEFINITION_SHAPE('name',$,#66);
#84= CARTESIAN_POINT('0',(0.,0.,0.));
#86= DUMMY_NODE('0',(#84),#36,#77);
#88= CARTESIAN_POINT('1',(0.,-2.,1.));
#90= NODE('1',(#88),#46,#77);
#91= SHAPE_DEFINITION_REPRESENTATION(#83,#92);
#92= POINT_REPRESENTATION('point_representation_name_1',(#93,#45),#46);
#93= NODE_SET('node_set_name_1',(#90,#99,#103,#107,#111,#115,#119,#123,
#127,#131,#135,#139,#143,#147,#151,#155,#159));
#97= CARTESIAN_POINT('2',(1.,-2.,1.));
#99= NODE('2',(#97),#46,#77);
#101= CARTESIAN_POINT('3',(2.,-2.,1.));
#103= NODE('3',(#101),#46,#77);
#105= CARTESIAN_POINT('4',(3.,-2.,1.));
#107= NODE('4',(#105),#46,#77);
#109= CARTESIAN_POINT('5',(4.,-2.,1.));
#111= NODE('5',(#109),#46,#77);
#113= CARTESIAN_POINT('6',(5.,-2.,1.));
#115= NODE('6',(#113),#46,#77);
#117= CARTESIAN_POINT('7',(6.,-2.,1.));
#119= NODE('7',(#117),#46,#77);
#121= CARTESIAN_POINT('8',(7.,-2.,1.));
#123= NODE('8',(#121),#46,#77);
#125= CARTESIAN_POINT('9',(8.,-2.,1.));
#127= NODE('9',(#125),#46,#77);
#129= CARTESIAN_POINT('10',(9.,-2.,1.));
#131= NODE('10',(#129),#46,#77);
#133= CARTESIAN_POINT('11',(10.,-2.,1.));
#135= NODE('11',(#133),#46,#77);
#137= CARTESIAN_POINT('12',(11.,-2.,1.));
#139= NODE('12',(#137),#46,#77);
#141= CARTESIAN_POINT('13',(12.,-2.,1.));
#143= NODE('13',(#141),#46,#77);
#145= CARTESIAN_POINT('14',(13.,-2.,1.));
#147= NODE('14',(#145),#46,#77);
#149= CARTESIAN_POINT('15',(14.,-2.,1.));
#151= NODE('15',(#149),#46,#77);
#153= CARTESIAN_POINT('16',(15.,-2.,1.));
#155= NODE('16',(#153),#46,#77);
#157= CARTESIAN_POINT('17',(16.,-2.,1.));
#159= NODE('17',(#157),#46,#77);
#161= ITEM_DEFINED_TRANSFORMATION('','',#35,#45);
#162= REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION('','',#77,#92,#161
```

```
);
#163= ELEMENT_MATERIAL('MAT1.1','Fea Material',(#175,#183,#189));
#164= MEASURE_REPRESENTATION_ITEM('representation_item_name_1',
THERMODYNAMIC_TEMPERATURE_MEASURE(70.),#10);
#165= PROPERTY_DEFINITION_REPRESENTATION(#166,#168);
#166= PROPERTY_DEFINITION('Material Property Definition',$,#167);
#167= CHARACTERIZED_OBJECT('property',$);
#168= REPRESENTATION('representation_id_1',(#164),#36);
#169= ID_ATTRIBUTE('MAT1.1',#166);
#171= DATA_ENVIRONMENT('DATA_ENV','Property_conditions',(#165));
#173= FEA_LINEAR_ELASTICITY('',FEA_ISOTROPIC_SYMMETRIC_TENSOR4_3D((
10000000.,0.33)));
#175= FEA_MATERIAL_PROPERTY_REPRESENTATION(#176,#178,#171);
#176= MATERIAL_PROPERTY('MAT1.1.0',$,#177);
#177= CHARACTERIZED_OBJECT('material_property',$);
#178= REPRESENTATION('representation_id_2',(#173),#36);
#179= DESCRIPTION_ATTRIBUTE('Elasticity',#175);
#182= FEA_MASS_DENSITY('',0.000254);
#183= FEA_MATERIAL_PROPERTY_REPRESENTATION(#184,#185,#171);
#184= MATERIAL_PROPERTY('MAT1.1.1',$,#177);
#185= REPRESENTATION('representation_id_3',(#182),#36);
#186= DESCRIPTION_ATTRIBUTE('MassDensity',#183);
#188= FEA_TANGENTIAL_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION('',
ISOTROPIC_SYMMETRIC_TENSOR2_3D(0.));
#189= FEA_MATERIAL_PROPERTY_REPRESENTATION(#190,#191,#171);
#190= MATERIAL_PROPERTY('MAT1.1.2',$,#177);
#191= REPRESENTATION('representation_id_4',(#188),#36);
#192= DESCRIPTION_ATTRIBUTE('TangentCTE',#189);
#194= CURVE_3D_ELEMENT_PROPERTY('PROD.1','',(#195),(#213,#216),(#218,
#222));
#195= CURVE_ELEMENT_INTERVAL_CONSTANT(#198,#196,#206);
#196= EULER_ANGLES((0.,0.,0.));
#198= CURVE_ELEMENT_LOCATION(#199);
#199= FEA_PARAMETRIC_POINT('',(1.,0.,0.));
#201= REPRESENTATION('dpr0',(#199),#202);
#202= PARAMETRIC_REPRESENTATION_CONTEXT('FEA parametric context',
'Parametric');
#206= CURVE_ELEMENT_SECTION_DERIVED_DEFINITIONS('',0.,8.,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),(0.,0.,0.),
0.,CONTEXT_DEPENDENT_MEASURE(0.),(CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),(CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),(CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.));
#213= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#216= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#218= CURVE_ELEMENT_END_RELEASE(#35,(#219));
#219= CURVE_ELEMENT_END_RELEASE_PACKET(ENUMERATED_CURVE_ELEMENT_FREEDOM(
.NONE.),0.);
#222= CURVE_ELEMENT_END_RELEASE(#35,(#219));
#224= CURVE_3D_ELEMENT_REPRESENTATION('1',(#233,#199),#225,(#90,#99),#77
,#238,#194,#163);
#225=
/* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CONTEXT()
REPRESENTATION_CONTEXT('REP_CONTEXT_FEA','3D'));
#227= REPRESENTATION('Default SimDM Fea representation',(#229),#225);
#229= REPRESENTATION_ITEM('Default SimDM item');
#230= ID_ATTRIBUTE('Default SimDM Fea representation',#227);
#231= DESCRIPTION_ATTRIBUTE('Default SimDM Fea representation',#227);
#233= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#236);
#234= DIRECTION('undefined',(1.,0.,0.));
#236= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#234);
#238= CURVE_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_CURVE.CROD',((
ENUMERATED_CURVE_ELEMENT_PURPOSE(.AXIAL.)),(
ENUMERATED_CURVE_ELEMENT_PURPOSE(.TORSION.))));
#242= CURVE_3D_ELEMENT_REPRESENTATION('2',(#233,#199),#225,(#99,#103),
#77,#238,#194,#163);
#245= CURVE_3D_ELEMENT_REPRESENTATION('3',(#233,#199),#225,(#103,#107),
#77,#238,#194,#163);
#248= CURVE_3D_ELEMENT_REPRESENTATION('4',(#233,#199),#225,(#107,#111),
```

```
#77,#238,#194,#163);
#251= CURVE_3D_ELEMENT_REPRESENTATION('5',(#233,#199),#225,(#111,#115),
#77,#238,#194,#163);
#254= CURVE_3D_ELEMENT_REPRESENTATION('6',(#233,#199),#225,(#115,#119),
#77,#238,#194,#163);
#257= CURVE_3D_ELEMENT_REPRESENTATION('7',(#233,#199),#225,(#119,#123),
#77,#238,#194,#163);
#260= CURVE_3D_ELEMENT_REPRESENTATION('8',(#233,#199),#225,(#123,#127),
#77,#238,#194,#163);
#263= CURVE_3D_ELEMENT_REPRESENTATION('9',(#233,#199),#225,(#127,#131),
#77,#238,#194,#163);
#266= CURVE_3D_ELEMENT_REPRESENTATION('10',(#233,#199),#225,(#131,#135),
#77,#238,#194,#163);
#269= CURVE_3D_ELEMENT_REPRESENTATION('11',(#233,#199),#225,(#135,#139),
#77,#238,#194,#163);
#272= CURVE_3D_ELEMENT_REPRESENTATION('12',(#233,#199),#225,(#139,#143),
#77,#238,#194,#163);
#275= CURVE_3D_ELEMENT_REPRESENTATION('13',(#233,#199),#225,(#143,#147),
#77,#238,#194,#163);
#278= CURVE_3D_ELEMENT_REPRESENTATION('14',(#233,#199),#225,(#147,#151),
#77,#238,#194,#163);
#281= CURVE_3D_ELEMENT_REPRESENTATION('15',(#233,#199),#225,(#151,#155),
#77,#238,#194,#163);
#284= CURVE_3D_ELEMENT_REPRESENTATION('16',(#233,#199),#225,(#155,#159),
#77,#238,#194,#163);
#287= CONTROL(#77,'Control.0','FemConvert',
'Nastran job EAS test case ATS1m4',('<SOL>101</SOL>','TIME 600','CEND',
'<GLOBALCASE>0</GLOBALCASE>','<SUBCASE>1</SUBCASE>','ENDDATA'),(
'NASTRAN'));
#290= SPECIFIED_STATE('STEP DEFAULT','default_initial_state');
#291= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#287,'STATIC STEP 1',1,#290,''
,#313);
#292= SPECIFIED_STATE('Step 1 Base Specified State',
'Relating Specified State');
#293= SPECIFIED_STATE('SPCCASE_100_200_1','Aggregator SPC 100 Step 1');
#294= SPECIFIED_STATE('SPCVALSTATE_100_200','');
#295= STATE_RELATIONSHIP(
'SPCVALSTATE_100_200 is related to SPCCASE_100_200_1','',#293,#294);
#296= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#294,#299,#297,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#297= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#299= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.0',(#291),#90,#35,(#301,
#304,#305),'');
#301= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.X_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#304= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.Y_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#305= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.Z_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#306= STATE_RELATIONSHIP(
'SPCCASE_100_200_1 is related to Step 1 Base Specified State','',#292,
#293);
#307= SPECIFIED_STATE('LOADSTATECORE_200','Core Loads Specified State');
#308= NODAL_FREEDOM_ACTION_DEFINITION(#307,#159,#35,#309,(
CONTEXT_DEPENDENT_MEASURE(-1000.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#309= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#312= STATE_RELATIONSHIP(
'LOADSTATECORE_200 is related to Step 1 Base Specified State','',#292,
#307);
#313= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 1','',
#292);
#314= OUTPUT_REQUEST_STATE('Step 1 Output Request State','',(#291));
#316= NODAL_FREEDOM_VALUES(#314,#320,#35,#317,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
```

```
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#317= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(
.Y_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTATION.)));
#320= NODE_GROUP('','ALL',#77,(#90,#99,#103,#107,#111,#115,#119,#123,
#127,#131,#135,#139,#143,#147,#151,#155,#159));
#321= ID_ATTRIBUTE('ALL',#320);
#323= NODAL_FREEDOM_ACTION_DEFINITION(#307,#159,#35,#309,(
CONTEXT_DEPENDENT_MEASURE(-1000.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
ENDSEC;

END-ISO-10303-21;
```

11.2 ATS2m4.stp

```
ISO-10303-21;
HEADER;

/***********************************************************************************************
* Creation date:                  Tue Feb 07 13:53:29 2017
* Schema:                         AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF
***********************************************************************************************/

FILE_DESCRIPTION((''),'2;1');
FILE_NAME(
'O:\\projects\\Crystal\\software_FEM\\converters\\examples\\db\\e.DataRepository.Fem_Model'

,
          '2017-02-07T13:53:29',
          ('ANONYMOUS USER'),
          ('ANONYMOUS ORGANISATION'),
          'EDMsix Version 2.0100.a14 Jan 31 2017',
          '',
          '');
FILE_SCHEMA(('AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF'));
ENDSEC;

DATA;
#1= APPLICATION_CONTEXT(
'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF');
#3= ID_ATTRIBUTE('to be decided',#1);
#4= DESCRIPTION_ATTRIBUTE('to be decided',#1);
#5= APPLICATION_PROTOCOL_DEFINITION('international standard',
'ap209_multidisciplinary_analysis_and_design',2011,#1);
#6= DIMENSIONAL_EXPONENTS(0.,0.,0.,0.,0.,0.,0.);
#7= /* LENGTH_UNIT+SI_UNIT */(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT($,
.METRE.));
#8= /* MASS_UNIT+SI_UNIT */(MASS_UNIT()NAMED_UNIT(*)SI_UNIT(.KILO.,
.GRAM.));
#9= /* SI_UNIT+TIME_UNIT */(NAMED_UNIT(*)SI_UNIT($,.SECOND.)TIME_UNIT()
);
#10= /* SI_UNIT+THERMODYNAMIC_TEMPERATURE_UNIT */(NAMED_UNIT(*)SI_UNIT($
,.DEGREE_CELSIUS.)THERMODYNAMIC_TEMPERATURE_UNIT());
#11= /* PLANE_ANGLE_UNIT+SI_UNIT */(NAMED_UNIT(*)PLANE_ANGLE_UNIT()
SI_UNIT($,.RADIAN.));
#12= SI_FORCE_UNIT((#13,#14,#15),*,$,.NEWTON.);
#13= DERIVED_UNIT_ELEMENT(#8,1.);
#14= DERIVED_UNIT_ELEMENT(#7,1.);
#15= DERIVED_UNIT_ELEMENT(#9,-2.);
#17= SI_FREQUENCY_UNIT((#18),*,$,.HERTZ.);
#18= DERIVED_UNIT_ELEMENT(#9,-1.);
#20= SI_PRESSURE_UNIT((#13,#21,#15),*,$,.PASCAL.);
#21= DERIVED_UNIT_ELEMENT(#7,-1.);
#23= SI_ENERGY_UNIT((#13,#24,#15),*,$,.JOULE.);
#24= DERIVED_UNIT_ELEMENT(#7,2.);
#26= SI_POWER_UNIT((#13,#24,#27),*,$,.WATT.);
#27= DERIVED_UNIT_ELEMENT(#9,-3.);
#29= DIRECTION('RefX',(1.,0.,0.));
#31= DIRECTION('AxisZ',(0.,0.,1.));
#33= CARTESIAN_POINT('',(0.,0.,0.));
#35= FEA_AXIS2_PLACEMENT_3D('0',#33,#31,#29,.CARTESIAN.,
'FEA_BASIC_COORD_SYSTEM.0');
#36= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT(
'FEA_BASIC_COORD_SYSTEM.0','3d'));
#39= DIRECTION('RefX',(1.,0.,0.));
#41= DIRECTION('AxisZ',(0.,0.,1.));
#43= CARTESIAN_POINT('',(0.,0.,0.));
#45= FEA_AXIS2_PLACEMENT_3D('1',#43,#41,#39,.CARTESIAN.,'CORD2R.1');
#46= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT('CORD2R.1','3d'
));
```

```
#49= ORGANIZATION($,'default-organization',$);
#50= PRODUCT('default-id.0','default-fea-part','',(#52));
#52= PRODUCT_CONTEXT('design_context',#1,'design_context');
#54= PRODUCT_RELATED_PRODUCT_CATEGORY('part',$,(#50));
#55= ID_ATTRIBUTE('',#54);
#57= APPLIED_IDENTIFICATION_ASSIGNMENT('default-id.0',#58,(#60,#66,#50)
);
#58= IDENTIFICATION_ROLE('default-role',$);
#59= PRODUCT_RELATED_PRODUCT_CATEGORY('product',$,(#50));
#60= PRODUCT_DEFINITION_FORMATION('default-id.0','',#50);
#61= GROUP('id',$);
#62= APPLIED_CLASSIFICATION_ASSIGNMENT(#61,#64,(#57));
#64= CLASSIFICATION_ROLE('',$);
#66= PRODUCT_DEFINITION('default-id.0',$,#60,#67);
#67= PRODUCT_DEFINITION_CONTEXT('assembly definition',#1,
'life cycle stage');
#68= APPLIED_CLASSIFICATION_ASSIGNMENT(#69,#70,(#66));
#69= GROUP('default-gpvv-type',$);
#70= CLASSIFICATION_ROLE('view_type',$);
#73= ORGANIZATION_ROLE('id context');
#74= APPLIED_ORGANIZATION_ASSIGNMENT(#49,#73,(#57));
#76= NAME_ATTRIBUTE('',#66);
#77= FEA_MODEL_3D('Identification',(#35),#36,
'NASTRAN BDF Converter v0.0.0',('NASTRAN'),'AnalysisModelType');
#80= STRUCTURAL_RESPONSE_PROPERTY_DEFINITION_REPRESENTATION(#81,#77);
#81= STRUCTURAL_RESPONSE_PROPERTY('structural_response_property_id_1',$,
#82);
#82= FEA_MODEL_DEFINITION('fea_model_definition_id_1',$,#83,.U.);
#83= PRODUCT_DEFINITION_SHAPE('name',$,#66);
#84= CARTESIAN_POINT('0',(0.,0.,0.));
#86= DUMMY_NODE('0',(#84),#36,#77);
#88= CARTESIAN_POINT('1',(0.,-2.,1.));
#90= NODE('1',(#88),#46,#77);
#91= SHAPE_DEFINITION_REPRESENTATION(#83,#92);
#92= POINT_REPRESENTATION('point_representation_name_1',(#93,#45),#46);
#93= NODE_SET('node_set_name_1',(#90,#99,#103,#107,#111,#115,#119,#123,
#127,#131,#135,#139,#143,#147,#151,#155,#159));
#97= CARTESIAN_POINT('2',(1.,-2.,1.));
#99= NODE('2',(#97),#46,#77);
#101= CARTESIAN_POINT('3',(2.,-2.,1.));
#103= NODE('3',(#101),#46,#77);
#105= CARTESIAN_POINT('4',(3.,-2.,1.));
#107= NODE('4',(#105),#46,#77);
#109= CARTESIAN_POINT('5',(4.,-2.,1.));
#111= NODE('5',(#109),#46,#77);
#113= CARTESIAN_POINT('6',(5.,-2.,1.));
#115= NODE('6',(#113),#46,#77);
#117= CARTESIAN_POINT('7',(6.,-2.,1.));
#119= NODE('7',(#117),#46,#77);
#121= CARTESIAN_POINT('8',(7.,-2.,1.));
#123= NODE('8',(#121),#46,#77);
#125= CARTESIAN_POINT('9',(8.,-2.,1.));
#127= NODE('9',(#125),#46,#77);
#129= CARTESIAN_POINT('10',(9.,-2.,1.));
#131= NODE('10',(#129),#46,#77);
#133= CARTESIAN_POINT('11',(10.,-2.,1.));
#135= NODE('11',(#133),#46,#77);
#137= CARTESIAN_POINT('12',(11.,-2.,1.));
#139= NODE('12',(#137),#46,#77);
#141= CARTESIAN_POINT('13',(12.,-2.,1.));
#143= NODE('13',(#141),#46,#77);
#145= CARTESIAN_POINT('14',(13.,-2.,1.));
#147= NODE('14',(#145),#46,#77);
#149= CARTESIAN_POINT('15',(14.,-2.,1.));
#151= NODE('15',(#149),#46,#77);
#153= CARTESIAN_POINT('16',(15.,-2.,1.));
#155= NODE('16',(#153),#46,#77);
#157= CARTESIAN_POINT('17',(16.,-2.,1.));
#159= NODE('17',(#157),#46,#77);
#161= ITEM_DEFINED_TRANSFORMATION('','',#35,#45);
#162= REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION('','',#77,#92,#161
```

```
);
#163= ELEMENT_MATERIAL('MAT1.1','Fea Material',(#175,#183,#189));
#164= MEASURE_REPRESENTATION_ITEM('representation_item_name_1',
THERMODYNAMIC_TEMPERATURE_MEASURE(70.),#10);
#165= PROPERTY_DEFINITION_REPRESENTATION(#166,#168);
#166= PROPERTY_DEFINITION('Material Property Definition',$,#167);
#167= CHARACTERIZED_OBJECT('property',$);
#168= REPRESENTATION('representation_id_1',(#164),#36);
#169= ID_ATTRIBUTE('MAT1.1',#166);
#171= DATA_ENVIRONMENT('DATA_ENV','Property_conditions',(#165));
#173= FEA_LINEAR_ELASTICITY('',FEA_ISOTROPIC_SYMMETRIC_TENSOR4_3D((
10000000.,0.33)));
#175= FEA_MATERIAL_PROPERTY_REPRESENTATION(#176,#178,#171);
#176= MATERIAL_PROPERTY('MAT1.1.0',$,#177);
#177= CHARACTERIZED_OBJECT('material_property',$);
#178= REPRESENTATION('representation_id_2',(#173),#36);
#179= DESCRIPTION_ATTRIBUTE('Elasticity',#175);
#182= FEA_MASS_DENSITY('',0.000254);
#183= FEA_MATERIAL_PROPERTY_REPRESENTATION(#184,#185,#171);
#184= MATERIAL_PROPERTY('MAT1.1.1',$,#177);
#185= REPRESENTATION('representation_id_3',(#182),#36);
#186= DESCRIPTION_ATTRIBUTE('MassDensity',#183);
#188= FEA_TANGENTIAL_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION('',
ISOTROPIC_SYMMETRIC_TENSOR2_3D(0.));
#189= FEA_MATERIAL_PROPERTY_REPRESENTATION(#190,#191,#171);
#190= MATERIAL_PROPERTY('MAT1.1.2',$,#177);
#191= REPRESENTATION('representation_id_4',(#188),#36);
#192= DESCRIPTION_ATTRIBUTE('TangentCTE',#189);
#194= CURVE_3D_ELEMENT_REPRESENTATION('1',(#203,#213),#195,(#90,#99),#77
,#238,#208,#163);
#195=
/* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CONTEXT()
REPRESENTATION_CONTEXT('REP_CONTEXT_FEA','3D'));
#197= REPRESENTATION('Default SimDM Fea representation',(#199),#195);
#199= REPRESENTATION_ITEM('Default SimDM item');
#200= ID_ATTRIBUTE('Default SimDM Fea representation',#197);
#201= DESCRIPTION_ATTRIBUTE('Default SimDM Fea representation',#197);
#203= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#204);
#204= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#205);
#205= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#208= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.1','',(#209),(#227,#230),(#232,
#236));
#209= CURVE_ELEMENT_INTERVAL_CONSTANT(#212,#210,#220);
#210= EULER_ANGLES((0.,0.,0.));
#212= CURVE_ELEMENT_LOCATION(#213);
#213= FEA_PARAMETRIC_POINT('',(1.,0.,0.));
#215= REPRESENTATION('dpr0',(#213),#216);
#216= PARAMETRIC_REPRESENTATION_CONTEXT('FEA parametric context',
'Parametric');
#220= CURVE_ELEMENT_SECTION_DERIVED_DEFINITIONS('PBAR.1_0',0.,8.,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)),(2.667,
10.667,0.),0.,CONTEXT_DEPENDENT_MEASURE(0.),(CONTEXT_DEPENDENT_MEASURE(
0.),CONTEXT_DEPENDENT_MEASURE(0.)),(CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),(CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.));
#227= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#230= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#232= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#233= CURVE_ELEMENT_END_RELEASE_PACKET(ENUMERATED_CURVE_ELEMENT_FREEDOM(
.NONE.),0.);
#236= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#238= CURVE_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_CURVE.CBAR*',((
ENUMERATED_CURVE_ELEMENT_PURPOSE(.AXIAL.)),(
ENUMERATED_CURVE_ELEMENT_PURPOSE(.Y_Y_BENDING.),
ENUMERATED_CURVE_ELEMENT_PURPOSE(.Z_Z_BENDING.)),(
ENUMERATED_CURVE_ELEMENT_PURPOSE(.TORSION.)),(
ENUMERATED_CURVE_ELEMENT_PURPOSE(.X_Y_SHEAR.),
ENUMERATED_CURVE_ELEMENT_PURPOSE(.X_Z_SHEAR.))));
#244= CURVE_3D_ELEMENT_REPRESENTATION('2',(#246,#213),#195,(#99,#103),
```

```
#77,#238,#251,#163);
#246= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#247);
#247= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#248);
#248= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#251= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.2','',(#252),(#257,#260),(#262,
#265));
#252= CURVE_ELEMENT_INTERVAL_CONSTANT(#255,#253,#220);
#253= EULER_ANGLES((0.,0.,0.));
#255= CURVE_ELEMENT_LOCATION(#213);
#257= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#260= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#262= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#265= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#267= CURVE_3D_ELEMENT_REPRESENTATION('3',(#269,#213),#195,(#103,#107),
#77,#238,#274,#163);
#269= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#270);
#270= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#271);
#271= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#274= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.3','',(#275),(#280,#283),(#285,
#288));
#275= CURVE_ELEMENT_INTERVAL_CONSTANT(#278,#276,#220);
#276= EULER_ANGLES((0.,0.,0.));
#278= CURVE_ELEMENT_LOCATION(#213);
#280= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#283= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#285= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#288= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#290= CURVE_3D_ELEMENT_REPRESENTATION('4',(#292,#213),#195,(#107,#111),
#77,#238,#297,#163);
#292= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#293);
#293= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#294);
#294= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#297= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.4','',(#298),(#303,#306),(#308,
#311));
#298= CURVE_ELEMENT_INTERVAL_CONSTANT(#301,#299,#220);
#299= EULER_ANGLES((0.,0.,0.));
#301= CURVE_ELEMENT_LOCATION(#213);
#303= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#306= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#308= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#311= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#313= CURVE_3D_ELEMENT_REPRESENTATION('5',(#315,#213),#195,(#111,#115),
#77,#238,#320,#163);
#315= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#316);
#316= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#317);
#317= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#320= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.5','',(#321),(#326,#329),(#331,
#334));
#321= CURVE_ELEMENT_INTERVAL_CONSTANT(#324,#322,#220);
#322= EULER_ANGLES((0.,0.,0.));
#324= CURVE_ELEMENT_LOCATION(#213);
#326= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#329= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#331= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#334= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#336= CURVE_3D_ELEMENT_REPRESENTATION('6',(#338,#213),#195,(#115,#119),
#77,#238,#343,#163);
#338= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#339);
#339= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#340);
#340= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#343= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.6','',(#344),(#349,#352),(#354,
#357));
#344= CURVE_ELEMENT_INTERVAL_CONSTANT(#347,#345,#220);
#345= EULER_ANGLES((0.,0.,0.));
#347= CURVE_ELEMENT_LOCATION(#213);
#349= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#352= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#354= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#357= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#359= CURVE_3D_ELEMENT_REPRESENTATION('7',(#361,#213),#195,(#119,#123),
#77,#238,#366,#163);
```

```
#361= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#362);
#362= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#363);
#363= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#366= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.7','',(#367),(#372,#375),(#377,
#380));
#367= CURVE_ELEMENT_INTERVAL_CONSTANT(#370,#368,#220);
#368= EULER_ANGLES((0.,0.,0.));
#370= CURVE_ELEMENT_LOCATION(#213);
#372= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#375= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#377= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#380= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#382= CURVE_3D_ELEMENT_REPRESENTATION('8',(#384,#213),#195,(#123,#127),
#77,#238,#389,#163);
#384= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#385);
#385= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#386);
#386= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#389= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.8','',(#390),(#395,#398),(#400,
#403));
#390= CURVE_ELEMENT_INTERVAL_CONSTANT(#393,#391,#220);
#391= EULER_ANGLES((0.,0.,0.));
#393= CURVE_ELEMENT_LOCATION(#213);
#395= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#398= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#400= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#403= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#405= CURVE_3D_ELEMENT_REPRESENTATION('9',(#407,#213),#195,(#127,#131),
#77,#238,#412,#163);
#407= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#408);
#408= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#409);
#409= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#412= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.9','',(#413),(#418,#421),(#423,
#426));
#413= CURVE_ELEMENT_INTERVAL_CONSTANT(#416,#414,#220);
#414= EULER_ANGLES((0.,0.,0.));
#416= CURVE_ELEMENT_LOCATION(#213);
#418= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#421= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#423= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#426= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#428= CURVE_3D_ELEMENT_REPRESENTATION('10',(#430,#213),#195,(#131,#135),
#77,#238,#435,#163);
#430= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#431);
#431= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#432);
#432= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#435= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.10','',(#436),(#441,#444),(#446,
#449));
#436= CURVE_ELEMENT_INTERVAL_CONSTANT(#439,#437,#220);
#437= EULER_ANGLES((0.,0.,0.));
#439= CURVE_ELEMENT_LOCATION(#213);
#441= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#444= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#446= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#449= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#451= CURVE_3D_ELEMENT_REPRESENTATION('11',(#453,#213),#195,(#135,#139),
#77,#238,#458,#163);
#453= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#454);
#454= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#455);
#455= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#458= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.11','',(#459),(#464,#467),(#469,
#472));
#459= CURVE_ELEMENT_INTERVAL_CONSTANT(#462,#460,#220);
#460= EULER_ANGLES((0.,0.,0.));
#462= CURVE_ELEMENT_LOCATION(#213);
#464= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#467= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#469= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#472= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#474= CURVE_3D_ELEMENT_REPRESENTATION('12',(#476,#213),#195,(#139,#143),
#77,#238,#481,#163);
#476= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#477);
```

```
#477= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#478);
#478= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#481= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.12','',(#482),(#487,#490),(#492,
#495));
#482= CURVE_ELEMENT_INTERVAL_CONSTANT(#485,#483,#220);
#483= EULER_ANGLES((0.,0.,0.));
#485= CURVE_ELEMENT_LOCATION(#213);
#487= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#490= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#492= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#495= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#497= CURVE_3D_ELEMENT_REPRESENTATION('13',(#499,#213),#195,(#143,#147),
#77,#238,#504,#163);
#499= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#500);
#500= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#501);
#501= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#504= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.13','',(#505),(#510,#513),(#515,
#518));
#505= CURVE_ELEMENT_INTERVAL_CONSTANT(#508,#506,#220);
#506= EULER_ANGLES((0.,0.,0.));
#508= CURVE_ELEMENT_LOCATION(#213);
#510= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#513= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#515= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#518= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#520= CURVE_3D_ELEMENT_REPRESENTATION('14',(#522,#213),#195,(#147,#151),
#77,#238,#527,#163);
#522= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#523);
#523= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#524);
#524= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#527= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.14','',(#528),(#533,#536),(#538,
#541));
#528= CURVE_ELEMENT_INTERVAL_CONSTANT(#531,#529,#220);
#529= EULER_ANGLES((0.,0.,0.));
#531= CURVE_ELEMENT_LOCATION(#213);
#533= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#536= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#538= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#541= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#543= CURVE_3D_ELEMENT_REPRESENTATION('15',(#545,#213),#195,(#151,#155),
#77,#238,#550,#163);
#545= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#546);
#546= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#547);
#547= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#550= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.15','',(#551),(#556,#559),(#561,
#564));
#551= CURVE_ELEMENT_INTERVAL_CONSTANT(#554,#552,#220);
#552= EULER_ANGLES((0.,0.,0.));
#554= CURVE_ELEMENT_LOCATION(#213);
#556= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#559= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#561= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#564= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#566= CURVE_3D_ELEMENT_REPRESENTATION('16',(#568,#213),#195,(#155,#159),
#77,#238,#573,#163);
#568= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_SYSTEM('',#569);
#569= PARAMETRIC_CURVE_3D_ELEMENT_COORDINATE_DIRECTION('',#570);
#570= DIRECTION('dxy',(0.,7.549790000000000E-8,1.));
#573= CURVE_3D_ELEMENT_PROPERTY('PBAR.1.16','',(#574),(#579,#582),(#584,
#587));
#574= CURVE_ELEMENT_INTERVAL_CONSTANT(#577,#575,#220);
#575= EULER_ANGLES((0.,0.,0.));
#577= CURVE_ELEMENT_LOCATION(#213);
#579= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#582= CURVE_ELEMENT_END_OFFSET(#35,(0.,0.,0.));
#584= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#587= CURVE_ELEMENT_END_RELEASE(#35,(#233));
#589= CONTROL(#77,'Control.0','FemConvert',
'Nastran EAS test case ATS2m4',('<SOL>101</SOL>','TIME 600','CEND',
'<GLOBALCASE>0</GLOBALCASE>','<SUBCASE>1</SUBCASE>',
'<SUBCASE>2</SUBCASE>','<SUBCASE>3</SUBCASE>','ENDDATA'),('NASTRAN'));
```

```
#592= SPECIFIED_STATE('STEP DEFAULT','default_initial_state');
#593= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#589,'STATIC STEP 1',1,#592,''
,#618);
#594= SPECIFIED_STATE('Step 1 Base Specified State',
'Relating Specified State');
#595= SPECIFIED_STATE('SPCCASE_100_200_1','Aggregator SPC 100 Step 1');
#596= SPECIFIED_STATE('SPCVALSTATE_100_200','');
#597= STATE_RELATIONSHIP(
'SPCVALSTATE_100_200 is related to SPCCASE_100_200_1','',#595,#596);
#598= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#596,#601,#599,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#599= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(
.Y_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTATION.)));
#601= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.0',(#593),#90,#35,(#603,
#606,#607,#608,#609,#610),'');
#603= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.X_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#606= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.Y_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#607= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.Z_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#608= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.)
,CONTEXT_DEPENDENT_MEASURE(1.));
#609= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.Y_ROTATION.)
,CONTEXT_DEPENDENT_MEASURE(1.));
#610= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTATION.)
,CONTEXT_DEPENDENT_MEASURE(1.));
#611= STATE_RELATIONSHIP(
'SPCCASE_100_200_1 is related to Step 1 Base Specified State','',#594,
#595);
#612= SPECIFIED_STATE('LOADSTATECORE_200','Core Loads Specified State');
#613= NODAL_FREEDOM_ACTION_DEFINITION(#612,#159,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(-1000.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#614= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#617= STATE_RELATIONSHIP(
'LOADSTATECORE_200 is related to Step 1 Base Specified State','',#594,
#612);
#618= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 1','',
#594);
#619= OUTPUT_REQUEST_STATE('Step 1 Output Request State','',(#593));
#621= NODAL_FREEDOM_VALUES(#619,#625,#35,#622,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#622= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(
.Y_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTATION.)));
#625= NODE_GROUP('','ALL',#77,(#90,#99,#103,#107,#111,#115,#119,#123,
#127,#131,#135,#139,#143,#147,#151,#155,#159));
#626= ID_ATTRIBUTE('ALL',#625);
#628= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#589,'STATIC STEP 2',2,#592,''
,#670);
#629= SPECIFIED_STATE('Step 2 Base Specified State',
'Relating Specified State');
#630= SPECIFIED_STATE('SPCCASE_10_22_2','Aggregator SPC 10 Step 2');
#631= SPECIFIED_STATE('SPCVALSTATE_100_22','');
#632= STATE_RELATIONSHIP(
'SPCVALSTATE_100_22 is related to SPCCASE_10_22_2','',#630,#631);
#633= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#631,#634,#599,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
```

```
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#634= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.1',(#628),#90,#35,(#603,
#606,#607,#608,#609,#610),'');
#638= STATE_RELATIONSHIP(
'SPCCASE_10_22_2 is related to Step 2 Base Specified State','',#629,#630
);
#639= LINEARLY_SUPERIMPOSED_STATE('LOADSTATECOMBINATION_22',
'Overall Factor Combined State');
#641= STATE_COMPONENT('OverallComp','',#639,1.);
#643= SPECIFIED_STATE('LOADSTATECORE_300','Core Loads Specified State');
#644= NODAL_FREEDOM_ACTION_DEFINITION(#643,#135,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#646= NODAL_FREEDOM_ACTION_DEFINITION(#643,#159,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#648= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_300',
'Item Factor State');
#649= STATE_RELATIONSHIP('LOADSTATEITEM_300 is related to OverallComp',
'',#641,#648);
#650= STATE_COMPONENT('ItemComp_22_300','',#648,1.);
#652= STATE_RELATIONSHIP(
'LOADSTATECORE_300 is related to ItemComp_22_300','',#650,#643);
#653= SPECIFIED_STATE('LOADSTATECORE_400','Core Loads Specified State');
#654= NODAL_FREEDOM_ACTION_DEFINITION(#653,#139,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#656= NODAL_FREEDOM_ACTION_DEFINITION(#653,#143,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#658= NODAL_FREEDOM_ACTION_DEFINITION(#653,#147,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#660= NODAL_FREEDOM_ACTION_DEFINITION(#653,#151,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#662= NODAL_FREEDOM_ACTION_DEFINITION(#653,#155,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#664= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_400',
'Item Factor State');
#665= STATE_RELATIONSHIP('LOADSTATEITEM_400 is related to OverallComp',
'',#641,#664);
#666= STATE_COMPONENT('ItemComp_22_400','',#664,1.);
#668= STATE_RELATIONSHIP(
'LOADSTATECORE_400 is related to ItemComp_22_400','',#666,#653);
#669= STATE_RELATIONSHIP(
'LOADSTATECOMBINATION_22 is related to Step 2 Base Specified State','',
#629,#639);
#670= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 2','',
#629);
#671= OUTPUT_REQUEST_STATE('Step 2 Output Request State','',(#628));
#673= NODAL_FREEDOM_VALUES(#671,#625,#35,#622,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#675= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#589,'STATIC STEP 3',3,#592,''
,#701);
#676= SPECIFIED_STATE('Step 3 Base Specified State',
'Relating Specified State');
#677= SPECIFIED_STATE('SPCCASE_10_23_3','Aggregator SPC 10 Step 3');
#678= SPECIFIED_STATE('SPCVALSTATE_100_23','');
#679= STATE_RELATIONSHIP(
'SPCVALSTATE_100_23 is related to SPCCASE_10_23_3','',#677,#678);
#680= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#678,#681,#599,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#681= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.2',(#675),#90,#35,(#603,
#606,#607,#608,#609,#610),'');
#685= STATE_RELATIONSHIP(
```

```
'SPCCASE_10_23_3 is related to Step 3 Base Specified State','',#676,#677
);
#686= LINEARLY_SUPERIMPOSED_STATE('LOADSTATECOMBINATION_23',
'Overall Factor Combined State');
#687= STATE_COMPONENT('OverallComp','',#686,1.);
#689= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_200',
'Item Factor State');
#690= STATE_RELATIONSHIP('LOADSTATEITEM_200 is related to OverallComp',
'',#687,#689);
#691= STATE_COMPONENT('ItemComp_23_200','',#689,1.);
#693= STATE_RELATIONSHIP(
'LOADSTATECORE_200 is related to ItemComp_23_200','',#691,#612);
#694= STATE_RELATIONSHIP('LOADSTATEITEM_300 is related to OverallComp',
'',#687,#648);
#695= STATE_COMPONENT('ItemComp_23_300','',#648,1.);
#696= STATE_RELATIONSHIP(
'LOADSTATECORE_300 is related to ItemComp_23_300','',#695,#643);
#697= STATE_RELATIONSHIP('LOADSTATEITEM_400 is related to OverallComp',
'',#687,#664);
#698= STATE_COMPONENT('ItemComp_23_400','',#664,1.);
#699= STATE_RELATIONSHIP(
'LOADSTATECORE_400 is related to ItemComp_23_400','',#698,#653);
#700= STATE_RELATIONSHIP(
'LOADSTATECOMBINATION_23 is related to Step 3 Base Specified State','',
#676,#686);
#701= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 3','',
#676);
#702= OUTPUT_REQUEST_STATE('Step 3 Output Request State','',(#675));
#704= NODAL_FREEDOM_VALUES(#702,#625,#35,#622,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#706= NODAL_FREEDOM_ACTION_DEFINITION(#612,#159,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(-1000.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#708= NODAL_FREEDOM_ACTION_DEFINITION(#643,#135,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#710= NODAL_FREEDOM_ACTION_DEFINITION(#643,#159,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#712= NODAL_FREEDOM_ACTION_DEFINITION(#653,#139,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#714= NODAL_FREEDOM_ACTION_DEFINITION(#653,#143,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#716= NODAL_FREEDOM_ACTION_DEFINITION(#653,#147,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#718= NODAL_FREEDOM_ACTION_DEFINITION(#653,#151,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#720= NODAL_FREEDOM_ACTION_DEFINITION(#653,#155,#35,#614,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
ENDSEC;

END-ISO-10303-21;
```

11.3 ATS3m4.stp

```
ISO-10303-21;
HEADER;

/*******************************************************************************************
* Creation date:                  Tue Feb 07 13:53:31 2017
* Schema:                         AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF
*******************************************************************************************/

FILE_DESCRIPTION((''),'2;1');
FILE_NAME(
'O:\\projects\\Crystal\\software_FEM\\converters\\examples\\db\\e.DataRepository.Fem_Model'

,
          '2017-02-07T13:53:31',
          ('ANONYMOUS USER'),
          ('ANONYMOUS ORGANISATION'),
          'EDMsix Version 2.0100.a14 Jan 31 2017',
          '',
          '');
FILE_SCHEMA(('AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF'));
ENDSEC;

DATA;
#1= APPLICATION_CONTEXT(
'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF');
#3= ID_ATTRIBUTE('to be decided',#1);
#4= DESCRIPTION_ATTRIBUTE('to be decided',#1);
#5= APPLICATION_PROTOCOL_DEFINITION('international standard',
'ap209_multidisciplinary_analysis_and_design',2011,#1);
#6= DIMENSIONAL_EXPONENTS(0.,0.,0.,0.,0.,0.,0.);
#7= /* LENGTH_UNIT+SI_UNIT */(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT($,
.METRE.));
#8= /* MASS_UNIT+SI_UNIT */(MASS_UNIT()NAMED_UNIT(*)SI_UNIT(.KILO.,
.GRAM.));
#9= /* SI_UNIT+TIME_UNIT */(NAMED_UNIT(*)SI_UNIT($,.SECOND.)TIME_UNIT()
);
#10= /* SI_UNIT+THERMODYNAMIC_TEMPERATURE_UNIT */(NAMED_UNIT(*)SI_UNIT($
,.DEGREE_CELSIUS.)THERMODYNAMIC_TEMPERATURE_UNIT());
#11= /* PLANE_ANGLE_UNIT+SI_UNIT */(NAMED_UNIT(*)PLANE_ANGLE_UNIT()
SI_UNIT($,.RADIAN.));
#12= SI_FORCE_UNIT((#13,#14,#15),*,$,.NEWTON.);
#13= DERIVED_UNIT_ELEMENT(#8,1.);
#14= DERIVED_UNIT_ELEMENT(#7,1.);
#15= DERIVED_UNIT_ELEMENT(#9,-2.);
#17= SI_FREQUENCY_UNIT((#18),*,$,.HERTZ.);
#18= DERIVED_UNIT_ELEMENT(#9,-1.);
#20= SI_PRESSURE_UNIT((#13,#21,#15),*,$,.PASCAL.);
#21= DERIVED_UNIT_ELEMENT(#7,-1.);
#23= SI_ENERGY_UNIT((#13,#24,#15),*,$,.JOULE.);
#24= DERIVED_UNIT_ELEMENT(#7,2.);
#26= SI_POWER_UNIT((#13,#24,#27),*,$,.WATT.);
#27= DERIVED_UNIT_ELEMENT(#9,-3.);
#29= DIRECTION('RefX',(1.,0.,0.));
#31= DIRECTION('AxisZ',(0.,0.,1.));
#33= CARTESIAN_POINT('',(0.,0.,0.));
#35= FEA_AXIS2_PLACEMENT_3D('0',#33,#31,#29,.CARTESIAN.,
'FEA_BASIC_COORD_SYSTEM.0');
#36= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT(
'FEA_BASIC_COORD_SYSTEM.0','3d'));
#39= ORGANIZATION($,'default-organization',$);
#40= PRODUCT('default-id.0','default-fea-part','',(#42));
#42= PRODUCT_CONTEXT('design_context',#1,'design_context');
#44= PRODUCT_RELATED_PRODUCT_CATEGORY('part',$,(#40));
#45= ID_ATTRIBUTE('',#44);
#47= APPLIED_IDENTIFICATION_ASSIGNMENT('default-id.0',#48,(#50,#56,#40)
);
#48= IDENTIFICATION_ROLE('default-role',$);
```

```
#49= PRODUCT_RELATED_PRODUCT_CATEGORY('product',$,(#40));
#50= PRODUCT_DEFINITION_FORMATION('default-id.0','',#40);
#51= GROUP('id',$);
#52= APPLIED_CLASSIFICATION_ASSIGNMENT(#51,#54,(#47));
#54= CLASSIFICATION_ROLE('',$);
#56= PRODUCT_DEFINITION('default-id.0',$,#50,#57);
#57= PRODUCT_DEFINITION_CONTEXT('assembly definition',#1,
'life cycle stage');
#58= APPLIED_CLASSIFICATION_ASSIGNMENT(#59,#60,(#56));
#59= GROUP('default-gpvv-type',$);
#60= CLASSIFICATION_ROLE('view_type',$);
#63= ORGANIZATION_ROLE('id context');
#64= APPLIED_ORGANIZATION_ASSIGNMENT(#39,#63,(#47));
#66= NAME_ATTRIBUTE('',#56);
#67= FEA_MODEL_3D('Identification',(#35),#36,
'NASTRAN BDF Converter v0.0.0',('NASTRAN'),'AnalysisModelType');
#70= STRUCTURAL_RESPONSE_PROPERTY_DEFINITION_REPRESENTATION(#71,#67);
#71= STRUCTURAL_RESPONSE_PROPERTY('structural_response_property_id_1',$,
#72);
#72= FEA_MODEL_DEFINITION('fea_model_definition_id_1',$,#73,.U.);
#73= PRODUCT_DEFINITION_SHAPE('name',$,#56);
#74= CARTESIAN_POINT('0',(0.,0.,0.));
#76= DUMMY_NODE('0',(#74),#36,#67);
#78= CARTESIAN_POINT('1',(-6.240220000000000E-8,-4.,1.));
#80= NODE('1',(#78),#36,#67);
#81= SHAPE_DEFINITION_REPRESENTATION(#73,#82);
#82= POINT_REPRESENTATION('point_representation_name_1',(#83,#35),#36);
#83= NODE_SET('node_set_name_1',(#80,#89,#93,#97,#101,#105,#109,#113,
#117,#121,#125,#129,#133,#137,#141,#145,#149,#153,#157,#161,#165,#169,
#173,#177,#181,#185,#189,#193,#197,#201,#205,#209,#213,#217,#221,#225,
#229,#233,#237,#241,#245,#249,#253,#257,#261,#265,#269,#273,#277,#281,
#285,#289,#293,#297,#301,#305,#309,#313,#317,#321,#325,#329,#333,#337,
#341,#345,#349,#353,#357,#361,#365,#369,#373,#377,#381,#385,#389,#393,
#397,#401,#405,#409,#413,#417,#421));
#87= CARTESIAN_POINT('2',(1.,-4.,1.));
#89= NODE('2',(#87),#36,#67);
#91= CARTESIAN_POINT('3',(2.,-4.,1.));
#93= NODE('3',(#91),#36,#67);
#95= CARTESIAN_POINT('4',(3.,-4.,1.));
#97= NODE('4',(#95),#36,#67);
#99= CARTESIAN_POINT('5',(4.,-4.,1.));
#101= NODE('5',(#99),#36,#67);
#103= CARTESIAN_POINT('6',(5.,-4.,1.));
#105= NODE('6',(#103),#36,#67);
#107= CARTESIAN_POINT('7',(6.,-4.,1.));
#109= NODE('7',(#107),#36,#67);
#111= CARTESIAN_POINT('8',(7.,-4.,1.));
#113= NODE('8',(#111),#36,#67);
#115= CARTESIAN_POINT('9',(8.,-4.,1.));
#117= NODE('9',(#115),#36,#67);
#119= CARTESIAN_POINT('10',(9.,-4.,1.));
#121= NODE('10',(#119),#36,#67);
#123= CARTESIAN_POINT('11',(10.,-4.,1.));
#125= NODE('11',(#123),#36,#67);
#127= CARTESIAN_POINT('12',(-6.240220000000000E-8,-3.,1.));
#129= NODE('12',(#127),#36,#67);
#131= CARTESIAN_POINT('13',(1.,-3.,1.));
#133= NODE('13',(#131),#36,#67);
#135= CARTESIAN_POINT('14',(2.,-3.,1.));
#137= NODE('14',(#135),#36,#67);
#139= CARTESIAN_POINT('15',(3.,-3.,1.));
#141= NODE('15',(#139),#36,#67);
#143= CARTESIAN_POINT('16',(4.,-3.,1.));
#145= NODE('16',(#143),#36,#67);
#147= CARTESIAN_POINT('17',(5.,-3.,1.));
#149= NODE('17',(#147),#36,#67);
#151= CARTESIAN_POINT('18',(6.,-3.,1.));
#153= NODE('18',(#151),#36,#67);
#155= CARTESIAN_POINT('19',(7.,-3.,1.));
#157= NODE('19',(#155),#36,#67);
#159= CARTESIAN_POINT('20',(8.,-3.,1.));
```

```
#161= NODE('20',(#159),#36,#67);
#163= CARTESIAN_POINT('21',(9.,-3.,1.));
#165= NODE('21',(#163),#36,#67);
#167= CARTESIAN_POINT('22',(10.,-3.,1.));
#169= NODE('22',(#167),#36,#67);
#171= CARTESIAN_POINT('23',(-6.240220000000000E-8,-2.,1.));
#173= NODE('23',(#171),#36,#67);
#175= CARTESIAN_POINT('24',(1.,-2.,1.));
#177= NODE('24',(#175),#36,#67);
#179= CARTESIAN_POINT('25',(2.,-2.,1.));
#181= NODE('25',(#179),#36,#67);
#183= CARTESIAN_POINT('26',(3.,-2.,1.));
#185= NODE('26',(#183),#36,#67);
#187= CARTESIAN_POINT('27',(4.,-2.,1.));
#189= NODE('27',(#187),#36,#67);
#191= CARTESIAN_POINT('28',(5.,-2.,1.));
#193= NODE('28',(#191),#36,#67);
#195= CARTESIAN_POINT('29',(6.,-2.,1.));
#197= NODE('29',(#195),#36,#67);
#199= CARTESIAN_POINT('30',(7.,-2.,1.));
#201= NODE('30',(#199),#36,#67);
#203= CARTESIAN_POINT('31',(8.,-2.,1.));
#205= NODE('31',(#203),#36,#67);
#207= CARTESIAN_POINT('32',(9.,-2.,1.));
#209= NODE('32',(#207),#36,#67);
#211= CARTESIAN_POINT('33',(10.,-2.,1.));
#213= NODE('33',(#211),#36,#67);
#215= CARTESIAN_POINT('34',(-6.240220000000000E-8,-1.,1.));
#217= NODE('34',(#215),#36,#67);
#219= CARTESIAN_POINT('35',(1.,-1.,1.));
#221= NODE('35',(#219),#36,#67);
#223= CARTESIAN_POINT('36',(2.,-1.,1.));
#225= NODE('36',(#223),#36,#67);
#227= CARTESIAN_POINT('37',(3.,-1.,1.));
#229= NODE('37',(#227),#36,#67);
#231= CARTESIAN_POINT('38',(4.,-1.,1.));
#233= NODE('38',(#231),#36,#67);
#235= CARTESIAN_POINT('39',(5.,-1.,1.));
#237= NODE('39',(#235),#36,#67);
#239= CARTESIAN_POINT('40',(6.,-1.,1.));
#241= NODE('40',(#239),#36,#67);
#243= CARTESIAN_POINT('41',(7.,-1.,1.));
#245= NODE('41',(#243),#36,#67);
#247= CARTESIAN_POINT('42',(8.,-1.,1.));
#249= NODE('42',(#247),#36,#67);
#251= CARTESIAN_POINT('43',(9.,-1.,1.));
#253= NODE('43',(#251),#36,#67);
#255= CARTESIAN_POINT('44',(10.,-1.,1.));
#257= NODE('44',(#255),#36,#67);
#259= CARTESIAN_POINT('45',(-6.240220000000000E-8,-5.275770000000000E-8,
1.));
#261= NODE('45',(#259),#36,#67);
#263= CARTESIAN_POINT('46',(1.,-5.275770000000000E-8,1.));
#265= NODE('46',(#263),#36,#67);
#267= CARTESIAN_POINT('47',(2.,-5.275770000000000E-8,1.));
#269= NODE('47',(#267),#36,#67);
#271= CARTESIAN_POINT('48',(3.,-5.275770000000000E-8,1.));
#273= NODE('48',(#271),#36,#67);
#275= CARTESIAN_POINT('49',(4.,-5.275770000000000E-8,1.));
#277= NODE('49',(#275),#36,#67);
#279= CARTESIAN_POINT('50',(5.,-5.275770000000000E-8,1.));
#281= NODE('50',(#279),#36,#67);
#283= CARTESIAN_POINT('51',(6.,-5.275770000000000E-8,1.));
#285= NODE('51',(#283),#36,#67);
#287= CARTESIAN_POINT('52',(7.,-5.275770000000000E-8,1.));
#289= NODE('52',(#287),#36,#67);
#291= CARTESIAN_POINT('53',(8.,-5.275770000000000E-8,1.));
#293= NODE('53',(#291),#36,#67);
#295= CARTESIAN_POINT('54',(9.,-5.275770000000000E-8,1.));
#297= NODE('54',(#295),#36,#67);
#299= CARTESIAN_POINT('55',(10.,-5.275770000000000E-8,1.));
```

```
#301= NODE('55',(#299),#36,#67);
#303= CARTESIAN_POINT('57',(11.,-4.,1.));
#305= NODE('57',(#303),#36,#67);
#307= CARTESIAN_POINT('58',(12.,-4.,1.));
#309= NODE('58',(#307),#36,#67);
#311= CARTESIAN_POINT('59',(13.,-4.,1.));
#313= NODE('59',(#311),#36,#67);
#315= CARTESIAN_POINT('60',(14.,-4.,1.));
#317= NODE('60',(#315),#36,#67);
#319= CARTESIAN_POINT('61',(15.,-4.,1.));
#321= NODE('61',(#319),#36,#67);
#323= CARTESIAN_POINT('62',(16.,-4.,1.));
#325= NODE('62',(#323),#36,#67);
#327= CARTESIAN_POINT('64',(11.,-3.,1.));
#329= NODE('64',(#327),#36,#67);
#331= CARTESIAN_POINT('65',(12.,-3.,1.));
#333= NODE('65',(#331),#36,#67);
#335= CARTESIAN_POINT('66',(13.,-3.,1.));
#337= NODE('66',(#335),#36,#67);
#339= CARTESIAN_POINT('67',(14.,-3.,1.));
#341= NODE('67',(#339),#36,#67);
#343= CARTESIAN_POINT('68',(15.,-3.,1.));
#345= NODE('68',(#343),#36,#67);
#347= CARTESIAN_POINT('69',(16.,-3.,1.));
#349= NODE('69',(#347),#36,#67);
#351= CARTESIAN_POINT('71',(11.,-2.,1.));
#353= NODE('71',(#351),#36,#67);
#355= CARTESIAN_POINT('72',(12.,-2.,1.));
#357= NODE('72',(#355),#36,#67);
#359= CARTESIAN_POINT('73',(13.,-2.,1.));
#361= NODE('73',(#359),#36,#67);
#363= CARTESIAN_POINT('74',(14.,-2.,1.));
#365= NODE('74',(#363),#36,#67);
#367= CARTESIAN_POINT('75',(15.,-2.,1.));
#369= NODE('75',(#367),#36,#67);
#371= CARTESIAN_POINT('76',(16.,-2.,1.));
#373= NODE('76',(#371),#36,#67);
#375= CARTESIAN_POINT('78',(11.,-1.,1.));
#377= NODE('78',(#375),#36,#67);
#379= CARTESIAN_POINT('79',(12.,-1.,1.));
#381= NODE('79',(#379),#36,#67);
#383= CARTESIAN_POINT('80',(13.,-1.,1.));
#385= NODE('80',(#383),#36,#67);
#387= CARTESIAN_POINT('81',(14.,-1.,1.));
#389= NODE('81',(#387),#36,#67);
#391= CARTESIAN_POINT('82',(15.,-1.,1.));
#393= NODE('82',(#391),#36,#67);
#395= CARTESIAN_POINT('83',(16.,-1.,1.));
#397= NODE('83',(#395),#36,#67);
#399= CARTESIAN_POINT('85',(11.,-5.275770000000000E-8,1.));
#401= NODE('85',(#399),#36,#67);
#403= CARTESIAN_POINT('86',(12.,-5.275770000000000E-8,1.));
#405= NODE('86',(#403),#36,#67);
#407= CARTESIAN_POINT('87',(13.,-5.275770000000000E-8,1.));
#409= NODE('87',(#407),#36,#67);
#411= CARTESIAN_POINT('88',(14.,-5.275770000000000E-8,1.));
#413= NODE('88',(#411),#36,#67);
#415= CARTESIAN_POINT('89',(15.,-5.275770000000000E-8,1.));
#417= NODE('89',(#415),#36,#67);
#419= CARTESIAN_POINT('90',(16.,-5.275770000000000E-8,1.));
#421= NODE('90',(#419),#36,#67);
#423= ELEMENT_MATERIAL('MAT1.1','Fea Material',(#435,#443,#449));
#424= MEASURE_REPRESENTATION_ITEM('representation_item_name_1',
THERMODYNAMIC_TEMPERATURE_MEASURE(70.),#10);
#425= PROPERTY_DEFINITION_REPRESENTATION(#426,#428);
#426= PROPERTY_DEFINITION('Material Property Definition',$,#427);
#427= CHARACTERIZED_OBJECT('property',$);
#428= REPRESENTATION('representation_id_1',(#424),#36);
#429= ID_ATTRIBUTE('MAT1.1',#426);
#431= DATA_ENVIRONMENT('DATA_ENV','Property_conditions',(#425));
#433= FEA_LINEAR_ELASTICITY('',FEA_ISOTROPIC_SYMMETRIC_TENSOR4_3D((
```

```
10000000.,0.33)));
#435= FEA_MATERIAL_PROPERTY_REPRESENTATION(#436,#438,#431);
#436= MATERIAL_PROPERTY('MAT1.1.0',$,#437);
#437= CHARACTERIZED_OBJECT('material_property',$);
#438= REPRESENTATION('representation_id_2',(#433),#36);
#439= DESCRIPTION_ATTRIBUTE('Elasticity',#435);
#442= FEA_MASS_DENSITY('',0.000254);
#443= FEA_MATERIAL_PROPERTY_REPRESENTATION(#444,#445,#431);
#444= MATERIAL_PROPERTY('MAT1.1.1',$,#437);
#445= REPRESENTATION('representation_id_3',(#442),#36);
#446= DESCRIPTION_ATTRIBUTE('MassDensity',#443);
#448= FEA_TANGENTIAL_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION('',
ISOTROPIC_SYMMETRIC_TENSOR2_3D(0.));
#449= FEA_MATERIAL_PROPERTY_REPRESENTATION(#450,#451,#431);
#450= MATERIAL_PROPERTY('MAT1.1.2',$,#437);
#451= REPRESENTATION('representation_id_4',(#448),#36);
#452= DESCRIPTION_ATTRIBUTE('TangentCTE',#449);
#454= SURFACE_3D_ELEMENT_REPRESENTATION('1',(#463),#455,(#80,#89,#133,
#129),#67,#468,#465,#423);
#455=
/* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CONTEXT()
REPRESENTATION_CONTEXT('REP_CONTEXT_FEA','3D'));
#457= REPRESENTATION('Default SimDM Fea representation',(#459),#455);
#459= REPRESENTATION_ITEM('Default SimDM item');
#460= ID_ATTRIBUTE('Default SimDM Fea representation',#457);
#461= DESCRIPTION_ATTRIBUTE('Default SimDM Fea representation',#457);
#463= PARAMETRIC_SURFACE_3D_ELEMENT_COORDINATE_SYSTEM('',1,0.);
#465= SURFACE_ELEMENT_PROPERTY('PSHELL.1','',#466);
#466= SURFACE_SECTION_FIELD_CONSTANT(#467);
#467= UNIFORM_SURFACE_SECTION(UNSPECIFIED_VALUE(.UNSPECIFIED.),
UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),2.,
UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.));
#468= SURFACE_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,
'LINEAR_QUADRILATERAL.CQUAD4',((ENUMERATED_SURFACE_ELEMENT_PURPOSE(
.MEMBRANE_DIRECT.),ENUMERATED_SURFACE_ELEMENT_PURPOSE(.MEMBRANE_SHEAR.))
,(ENUMERATED_SURFACE_ELEMENT_PURPOSE(.BENDING_DIRECT.),
ENUMERATED_SURFACE_ELEMENT_PURPOSE(.BENDING_TORSION.)),(
ENUMERATED_SURFACE_ELEMENT_PURPOSE(.NORMAL_TO_PLANE_SHEAR.))),
.QUADRILATERAL.);
#473= SURFACE_3D_ELEMENT_REPRESENTATION('2',(#463),#455,(#89,#93,#137,
#133),#67,#468,#465,#423);
#476= SURFACE_3D_ELEMENT_REPRESENTATION('3',(#463),#455,(#93,#97,#141,
#137),#67,#468,#465,#423);
#479= SURFACE_3D_ELEMENT_REPRESENTATION('4',(#463),#455,(#97,#101,#145,
#141),#67,#468,#465,#423);
#482= SURFACE_3D_ELEMENT_REPRESENTATION('5',(#463),#455,(#101,#105,#149,
#145),#67,#468,#465,#423);
#485= SURFACE_3D_ELEMENT_REPRESENTATION('6',(#463),#455,(#105,#109,#153,
#149),#67,#468,#465,#423);
#488= SURFACE_3D_ELEMENT_REPRESENTATION('7',(#463),#455,(#109,#113,#157,
#153),#67,#468,#465,#423);
#491= SURFACE_3D_ELEMENT_REPRESENTATION('8',(#463),#455,(#113,#117,#161,
#157),#67,#468,#465,#423);
#494= SURFACE_3D_ELEMENT_REPRESENTATION('9',(#463),#455,(#117,#121,#165,
#161),#67,#468,#465,#423);
#497= SURFACE_3D_ELEMENT_REPRESENTATION('10',(#463),#455,(#121,#125,#169
,#165),#67,#468,#465,#423);
#500= SURFACE_3D_ELEMENT_REPRESENTATION('11',(#463),#455,(#129,#133,#177
,#173),#67,#468,#465,#423);
#503= SURFACE_3D_ELEMENT_REPRESENTATION('12',(#463),#455,(#133,#137,#181
,#177),#67,#468,#465,#423);
#506= SURFACE_3D_ELEMENT_REPRESENTATION('13',(#463),#455,(#137,#141,#185
,#181),#67,#468,#465,#423);
#509= SURFACE_3D_ELEMENT_REPRESENTATION('14',(#463),#455,(#141,#145,#189
,#185),#67,#468,#465,#423);
#512= SURFACE_3D_ELEMENT_REPRESENTATION('15',(#463),#455,(#145,#149,#193
,#189),#67,#468,#465,#423);
#515= SURFACE_3D_ELEMENT_REPRESENTATION('16',(#463),#455,(#149,#153,#197
,#193),#67,#468,#465,#423);
#518= SURFACE_3D_ELEMENT_REPRESENTATION('17',(#463),#455,(#153,#157,#201
```

```
,#197),#67,#468,#465,#423);
#521= SURFACE_3D_ELEMENT_REPRESENTATION('18',(#463),#455,(#157,#161,#205
,#201),#67,#468,#465,#423);
#524= SURFACE_3D_ELEMENT_REPRESENTATION('19',(#463),#455,(#161,#165,#209
,#205),#67,#468,#465,#423);
#527= SURFACE_3D_ELEMENT_REPRESENTATION('20',(#463),#455,(#165,#169,#213
,#209),#67,#468,#465,#423);
#530= SURFACE_3D_ELEMENT_REPRESENTATION('21',(#463),#455,(#173,#177,#221
,#217),#67,#468,#465,#423);
#533= SURFACE_3D_ELEMENT_REPRESENTATION('22',(#463),#455,(#177,#181,#225
,#221),#67,#468,#465,#423);
#536= SURFACE_3D_ELEMENT_REPRESENTATION('23',(#463),#455,(#181,#185,#229
,#225),#67,#468,#465,#423);
#539= SURFACE_3D_ELEMENT_REPRESENTATION('24',(#463),#455,(#185,#189,#233
,#229),#67,#468,#465,#423);
#542= SURFACE_3D_ELEMENT_REPRESENTATION('25',(#463),#455,(#189,#193,#237
,#233),#67,#468,#465,#423);
#545= SURFACE_3D_ELEMENT_REPRESENTATION('26',(#463),#455,(#193,#197,#241
,#237),#67,#468,#465,#423);
#548= SURFACE_3D_ELEMENT_REPRESENTATION('27',(#463),#455,(#197,#201,#245
,#241),#67,#468,#465,#423);
#551= SURFACE_3D_ELEMENT_REPRESENTATION('28',(#463),#455,(#201,#205,#249
,#245),#67,#468,#465,#423);
#554= SURFACE_3D_ELEMENT_REPRESENTATION('29',(#463),#455,(#205,#209,#253
,#249),#67,#468,#465,#423);
#557= SURFACE_3D_ELEMENT_REPRESENTATION('30',(#463),#455,(#209,#213,#257
,#253),#67,#468,#465,#423);
#560= SURFACE_3D_ELEMENT_REPRESENTATION('31',(#463),#455,(#217,#221,#265
,#261),#67,#468,#465,#423);
#563= SURFACE_3D_ELEMENT_REPRESENTATION('32',(#463),#455,(#221,#225,#269
,#265),#67,#468,#465,#423);
#566= SURFACE_3D_ELEMENT_REPRESENTATION('33',(#463),#455,(#225,#229,#273
,#269),#67,#468,#465,#423);
#569= SURFACE_3D_ELEMENT_REPRESENTATION('34',(#463),#455,(#229,#233,#277
,#273),#67,#468,#465,#423);
#572= SURFACE_3D_ELEMENT_REPRESENTATION('35',(#463),#455,(#233,#237,#281
,#277),#67,#468,#465,#423);
#575= SURFACE_3D_ELEMENT_REPRESENTATION('36',(#463),#455,(#237,#241,#285
,#281),#67,#468,#465,#423);
#578= SURFACE_3D_ELEMENT_REPRESENTATION('37',(#463),#455,(#241,#245,#289
,#285),#67,#468,#465,#423);
#581= SURFACE_3D_ELEMENT_REPRESENTATION('38',(#463),#455,(#245,#249,#293
,#289),#67,#468,#465,#423);
#584= SURFACE_3D_ELEMENT_REPRESENTATION('39',(#463),#455,(#249,#253,#297
,#293),#67,#468,#465,#423);
#587= SURFACE_3D_ELEMENT_REPRESENTATION('40',(#463),#455,(#253,#257,#301
,#297),#67,#468,#465,#423);
#590= SURFACE_3D_ELEMENT_REPRESENTATION('41',(#592),#455,(#329,#169,#125
),#67,#594,#465,#423);
#592= PARAMETRIC_SURFACE_3D_ELEMENT_COORDINATE_SYSTEM('',1,0.);
#594= SURFACE_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,
'LINEAR_TRIANGLE.CTRIA3',((ENUMERATED_SURFACE_ELEMENT_PURPOSE(
.MEMBRANE_DIRECT.),ENUMERATED_SURFACE_ELEMENT_PURPOSE(.MEMBRANE_SHEAR.))
,(ENUMERATED_SURFACE_ELEMENT_PURPOSE(.BENDING_DIRECT.),
ENUMERATED_SURFACE_ELEMENT_PURPOSE(.BENDING_TORSION.)),(
ENUMERATED_SURFACE_ELEMENT_PURPOSE(.NORMAL_TO_PLANE_SHEAR.))),.TRIANGLE.
);
#599= SURFACE_3D_ELEMENT_REPRESENTATION('42',(#592),#455,(#125,#305,#329
),#67,#594,#465,#423);
#602= SURFACE_3D_ELEMENT_REPRESENTATION('43',(#592),#455,(#305,#309,#329
),#67,#594,#465,#423);
#605= SURFACE_3D_ELEMENT_REPRESENTATION('44',(#592),#455,(#333,#329,#309
),#67,#594,#465,#423);
#608= SURFACE_3D_ELEMENT_REPRESENTATION('45',(#592),#455,(#337,#333,#309
),#67,#594,#465,#423);
#611= SURFACE_3D_ELEMENT_REPRESENTATION('46',(#592),#455,(#309,#313,#337
),#67,#594,#465,#423);
#614= SURFACE_3D_ELEMENT_REPRESENTATION('47',(#592),#455,(#313,#317,#337
),#67,#594,#465,#423);
#617= SURFACE_3D_ELEMENT_REPRESENTATION('48',(#592),#455,(#341,#337,#317
),#67,#594,#465,#423);
```

```
#620= SURFACE_3D_ELEMENT_REPRESENTATION('49',(#592),#455,(#345,#341,#317
),#67,#594,#465,#423);
#623= SURFACE_3D_ELEMENT_REPRESENTATION('50',(#592),#455,(#317,#321,#345
),#67,#594,#465,#423);
#626= SURFACE_3D_ELEMENT_REPRESENTATION('51',(#592),#455,(#321,#325,#345
),#67,#594,#465,#423);
#629= SURFACE_3D_ELEMENT_REPRESENTATION('52',(#592),#455,(#349,#345,#325
),#67,#594,#465,#423);
#632= SURFACE_3D_ELEMENT_REPRESENTATION('53',(#592),#455,(#169,#329,#213
),#67,#594,#465,#423);
#635= SURFACE_3D_ELEMENT_REPRESENTATION('54',(#592),#455,(#353,#213,#329
),#67,#594,#465,#423);
#638= SURFACE_3D_ELEMENT_REPRESENTATION('55',(#592),#455,(#357,#353,#329
),#67,#594,#465,#423);
#641= SURFACE_3D_ELEMENT_REPRESENTATION('56',(#592),#455,(#329,#333,#357
),#67,#594,#465,#423);
#644= SURFACE_3D_ELEMENT_REPRESENTATION('57',(#592),#455,(#333,#337,#357
),#67,#594,#465,#423);
#647= SURFACE_3D_ELEMENT_REPRESENTATION('58',(#592),#455,(#361,#357,#337
),#67,#594,#465,#423);
#650= SURFACE_3D_ELEMENT_REPRESENTATION('59',(#592),#455,(#365,#361,#337
),#67,#594,#465,#423);
#653= SURFACE_3D_ELEMENT_REPRESENTATION('60',(#592),#455,(#337,#341,#365
),#67,#594,#465,#423);
#656= SURFACE_3D_ELEMENT_REPRESENTATION('61',(#592),#455,(#341,#345,#365
),#67,#594,#465,#423);
#659= SURFACE_3D_ELEMENT_REPRESENTATION('62',(#592),#455,(#369,#365,#345
),#67,#594,#465,#423);
#662= SURFACE_3D_ELEMENT_REPRESENTATION('63',(#592),#455,(#373,#369,#345
),#67,#594,#465,#423);
#665= SURFACE_3D_ELEMENT_REPRESENTATION('64',(#592),#455,(#345,#349,#373
),#67,#594,#465,#423);
#668= SURFACE_3D_ELEMENT_REPRESENTATION('65',(#592),#455,(#377,#257,#213
),#67,#594,#465,#423);
#671= SURFACE_3D_ELEMENT_REPRESENTATION('66',(#592),#455,(#213,#353,#377
),#67,#594,#465,#423);
#674= SURFACE_3D_ELEMENT_REPRESENTATION('67',(#592),#455,(#353,#357,#377
),#67,#594,#465,#423);
#677= SURFACE_3D_ELEMENT_REPRESENTATION('68',(#592),#455,(#381,#377,#357
),#67,#594,#465,#423);
#680= SURFACE_3D_ELEMENT_REPRESENTATION('69',(#592),#455,(#385,#381,#357
),#67,#594,#465,#423);
#683= SURFACE_3D_ELEMENT_REPRESENTATION('70',(#592),#455,(#357,#361,#385
),#67,#594,#465,#423);
#686= SURFACE_3D_ELEMENT_REPRESENTATION('71',(#592),#455,(#361,#365,#385
),#67,#594,#465,#423);
#689= SURFACE_3D_ELEMENT_REPRESENTATION('72',(#592),#455,(#389,#385,#365
),#67,#594,#465,#423);
#692= SURFACE_3D_ELEMENT_REPRESENTATION('73',(#592),#455,(#393,#389,#365
),#67,#594,#465,#423);
#695= SURFACE_3D_ELEMENT_REPRESENTATION('74',(#592),#455,(#365,#369,#393
),#67,#594,#465,#423);
#698= SURFACE_3D_ELEMENT_REPRESENTATION('75',(#592),#455,(#369,#373,#393
),#67,#594,#465,#423);
#701= SURFACE_3D_ELEMENT_REPRESENTATION('76',(#592),#455,(#397,#393,#373
),#67,#594,#465,#423);
#704= SURFACE_3D_ELEMENT_REPRESENTATION('77',(#592),#455,(#257,#377,#301
),#67,#594,#465,#423);
#707= SURFACE_3D_ELEMENT_REPRESENTATION('78',(#592),#455,(#401,#301,#377
),#67,#594,#465,#423);
#710= SURFACE_3D_ELEMENT_REPRESENTATION('79',(#592),#455,(#405,#401,#377
),#67,#594,#465,#423);
#713= SURFACE_3D_ELEMENT_REPRESENTATION('80',(#592),#455,(#377,#381,#405
),#67,#594,#465,#423);
#716= SURFACE_3D_ELEMENT_REPRESENTATION('81',(#592),#455,(#381,#385,#405
),#67,#594,#465,#423);
#719= SURFACE_3D_ELEMENT_REPRESENTATION('82',(#592),#455,(#409,#405,#385
),#67,#594,#465,#423);
#722= SURFACE_3D_ELEMENT_REPRESENTATION('83',(#592),#455,(#413,#409,#385
),#67,#594,#465,#423);
#725= SURFACE_3D_ELEMENT_REPRESENTATION('84',(#592),#455,(#385,#389,#413
```

```
),#67,#594,#465,#423);
#728= SURFACE_3D_ELEMENT_REPRESENTATION('85',(#592),#455,(#389,#393,#413
),#67,#594,#465,#423);
#731= SURFACE_3D_ELEMENT_REPRESENTATION('86',(#592),#455,(#417,#413,#393
),#67,#594,#465,#423);
#734= SURFACE_3D_ELEMENT_REPRESENTATION('87',(#592),#455,(#421,#417,#393
),#67,#594,#465,#423);
#737= SURFACE_3D_ELEMENT_REPRESENTATION('88',(#592),#455,(#393,#397,#421
),#67,#594,#465,#423);
#740= CONTROL(#67,'Control.0','FemConvert',
'Nastran job EAS test case ATS3m4',('<SOL>101</SOL>','TIME 600','CEND',
'<GLOBALCASE>0</GLOBALCASE>','<SUBCASE>1</SUBCASE>',
'<SUBCASE>2</SUBCASE>','<SUBCASE>3</SUBCASE>','<SUBCASE>4</SUBCASE>',
'ENDDATA'),('NASTRAN'));
#743= SPECIFIED_STATE('STEP DEFAULT','default_initial_state');
#744= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#740,'STATIC STEP 1',1,#743,''
,#1238);
#745= SPECIFIED_STATE('Step 1 Base Specified State',
'Relating Specified State');
#746= SPECIFIED_STATE('SPCCASE_11_21_1','Aggregator SPC 11 Step 1');
#747= SPECIFIED_STATE('SPCVALSTATE_100_21','');
#748= STATE_RELATIONSHIP(
'SPCVALSTATE_100_21 is related to SPCCASE_11_21_1','',#746,#747);
#749= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#747,#752,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#750= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#752= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.0',(#744),#173,#35,(#754
,#757,#758),'');
#754= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.X_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#757= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.Y_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#758= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.Z_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#759= SPECIFIED_STATE('SPCVALSTATE_101_21','');
#760= STATE_RELATIONSHIP(
'SPCVALSTATE_101_21 is related to SPCCASE_11_21_1','',#746,#759);
#761= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#759,#764,#762,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#762= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.)));
#764= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.1',(#744),#80,#35,(#754)
,'');
#768= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#759,#769,#762,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#769= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.2',(#744),#129,#35,(#754
),'');
#773= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#759,#774,#762,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#774= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.3',(#744),#217,#35,(#754
),'');
#778= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#759,#779,#762,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#779= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.4',(#744),#261,#35,(#754
),'');
#783= SPECIFIED_STATE('SPCVALSTATE_110_21','');
#784= STATE_RELATIONSHIP(
'SPCVALSTATE_110_21 is related to SPCCASE_11_21_1','',#746,#783);
#785= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#788,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#786= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_ROTATION.)));
#788= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.5',(#744),#80,#35,(#790,
#793),'');
#790= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.)
,CONTEXT_DEPENDENT_MEASURE(1.));
#793= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(.Y_ROTATION.)
,CONTEXT_DEPENDENT_MEASURE(1.));
#794= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#795,#786,(
```

```
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#795= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.6',(#744),#89,#35,(#790,
#793),'');
#799= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#800,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#800= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.7',(#744),#93,#35,(#790,
#793),'');
#804= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#805,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#805= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.8',(#744),#97,#35,(#790,
#793),'');
#809= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#810,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#810= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.9',(#744),#101,#35,(#790
,#793),'');
#814= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#815,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#815= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.10',(#744),#105,#35,(
#790,#793),'');
#819= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#820,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#820= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.11',(#744),#109,#35,(
#790,#793),'');
#824= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#825,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#825= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.12',(#744),#113,#35,(
#790,#793),'');
#829= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#830,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#830= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.13',(#744),#117,#35,(
#790,#793),'');
#834= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#835,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#835= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.14',(#744),#121,#35,(
#790,#793),'');
#839= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#840,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#840= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.15',(#744),#125,#35,(
#790,#793),'');
#844= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#845,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#845= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.16',(#744),#129,#35,(
#790,#793),'');
#849= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#850,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#850= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.17',(#744),#133,#35,(
#790,#793),'');
#854= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#855,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#855= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.18',(#744),#137,#35,(
#790,#793),'');
#859= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#860,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#860= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.19',(#744),#141,#35,(
#790,#793),'');
#864= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#865,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#865= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.20',(#744),#145,#35,(
#790,#793),'');
#869= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#870,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#870= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.21',(#744),#149,#35,(
#790,#793),'');
#874= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#875,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#875= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.22',(#744),#153,#35,(
#790,#793),'');
#879= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#880,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#880= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.23',(#744),#157,#35,(
#790,#793),'');
```

```
#884= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#885,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#885= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.24',(#744),#161,#35,(
#790,#793),'');
#889= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#890,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#890= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.25',(#744),#165,#35,(
#790,#793),'');
#894= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#895,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#895= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.26',(#744),#169,#35,(
#790,#793),'');
#899= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#900,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#900= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.27',(#744),#173,#35,(
#790,#793),'');
#904= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#905,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#905= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.28',(#744),#177,#35,(
#790,#793),'');
#909= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#910,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#910= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.29',(#744),#181,#35,(
#790,#793),'');
#914= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#915,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#915= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.30',(#744),#185,#35,(
#790,#793),'');
#919= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#920,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#920= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.31',(#744),#189,#35,(
#790,#793),'');
#924= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#925,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#925= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.32',(#744),#193,#35,(
#790,#793),'');
#929= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#930,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#930= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.33',(#744),#197,#35,(
#790,#793),'');
#934= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#935,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#935= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.34',(#744),#201,#35,(
#790,#793),'');
#939= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#940,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#940= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.35',(#744),#205,#35,(
#790,#793),'');
#944= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#945,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#945= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.36',(#744),#209,#35,(
#790,#793),'');
#949= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#950,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#950= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.37',(#744),#213,#35,(
#790,#793),'');
#954= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#955,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#955= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.38',(#744),#217,#35,(
#790,#793),'');
#959= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#960,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#960= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.39',(#744),#221,#35,(
#790,#793),'');
#964= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#965,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#965= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.40',(#744),#225,#35,(
#790,#793),'');
#969= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#970,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#970= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.41',(#744),#229,#35,(
```

```
#790,#793),'');
#974= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#975,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#975= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.42',(#744),#233,#35,(
#790,#793),'');
#979= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#980,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#980= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.43',(#744),#237,#35,(
#790,#793),'');
#984= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#985,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#985= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.44',(#744),#241,#35,(
#790,#793),'');
#989= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#990,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#990= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.45',(#744),#245,#35,(
#790,#793),'');
#994= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#995,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#995= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.46',(#744),#249,#35,(
#790,#793),'');
#999= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1000,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1000= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.47',(#744),#253,#35,(
#790,#793),'');
#1004= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1005,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1005= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.48',(#744),#257,#35,(
#790,#793),'');
#1009= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1010,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1010= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.49',(#744),#261,#35,(
#790,#793),'');
#1014= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1015,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1015= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.50',(#744),#265,#35,(
#790,#793),'');
#1019= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1020,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1020= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.51',(#744),#269,#35,(
#790,#793),'');
#1024= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1025,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1025= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.52',(#744),#273,#35,(
#790,#793),'');
#1029= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1030,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1030= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.53',(#744),#277,#35,(
#790,#793),'');
#1034= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1035,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1035= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.54',(#744),#281,#35,(
#790,#793),'');
#1039= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1040,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1040= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.55',(#744),#285,#35,(
#790,#793),'');
#1044= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1045,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1045= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.56',(#744),#289,#35,(
#790,#793),'');
#1049= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1050,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1050= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.57',(#744),#293,#35,(
#790,#793),'');
#1054= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1055,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1055= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.58',(#744),#297,#35,(
#790,#793),'');
#1059= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1060,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
```

```
#1060= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.59',(#744),#301,#35,(
#790,#793),'');
#1064= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1065,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1065= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.60',(#744),#305,#35,(
#790,#793),'');
#1069= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1070,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1070= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.61',(#744),#309,#35,(
#790,#793),'');
#1074= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1075,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1075= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.62',(#744),#313,#35,(
#790,#793),'');
#1079= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1080,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1080= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.63',(#744),#317,#35,(
#790,#793),'');
#1084= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1085,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1085= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.64',(#744),#321,#35,(
#790,#793),'');
#1089= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1090,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1090= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.65',(#744),#325,#35,(
#790,#793),'');
#1094= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1095,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1095= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.66',(#744),#329,#35,(
#790,#793),'');
#1099= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1100,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1100= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.67',(#744),#333,#35,(
#790,#793),'');
#1104= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1105,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1105= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.68',(#744),#337,#35,(
#790,#793),'');
#1109= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1110,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1110= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.69',(#744),#341,#35,(
#790,#793),'');
#1114= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1115,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1115= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.70',(#744),#345,#35,(
#790,#793),'');
#1119= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1120,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1120= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.71',(#744),#349,#35,(
#790,#793),'');
#1124= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1125,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1125= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.72',(#744),#353,#35,(
#790,#793),'');
#1129= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1130,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1130= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.73',(#744),#357,#35,(
#790,#793),'');
#1134= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1135,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1135= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.74',(#744),#361,#35,(
#790,#793),'');
#1139= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1140,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1140= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.75',(#744),#365,#35,(
#790,#793),'');
#1144= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1145,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1145= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.76',(#744),#369,#35,(
#790,#793),'');
#1149= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1150,#786,(
```

```
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1150= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.77',(#744),#373,#35,(
#790,#793),'');
#1154= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1155,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1155= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.78',(#744),#377,#35,(
#790,#793),'');
#1159= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1160,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1160= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.79',(#744),#381,#35,(
#790,#793),'');
#1164= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1165,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1165= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.80',(#744),#385,#35,(
#790,#793),'');
#1169= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1170,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1170= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.81',(#744),#389,#35,(
#790,#793),'');
#1174= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1175,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1175= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.82',(#744),#393,#35,(
#790,#793),'');
#1179= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1180,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1180= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.83',(#744),#397,#35,(
#790,#793),'');
#1184= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1185,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1185= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.84',(#744),#401,#35,(
#790,#793),'');
#1189= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1190,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1190= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.85',(#744),#405,#35,(
#790,#793),'');
#1194= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1195,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1195= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.86',(#744),#409,#35,(
#790,#793),'');
#1199= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1200,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1200= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.87',(#744),#413,#35,(
#790,#793),'');
#1204= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1205,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1205= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.88',(#744),#417,#35,(
#790,#793),'');
#1209= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#783,#1210,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1210= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.89',(#744),#421,#35,(
#790,#793),'');
#1214= STATE_RELATIONSHIP(
'SPCCASE_11_21_1 is related to Step 1 Base Specified State','',#745,#746
);
#1215= LINEARLY_SUPERIMPOSED_STATE('LOADSTATECOMBINATION_21',
'Overall Factor Combined State');
#1217= STATE_COMPONENT('OverallComp','',#1215,1.);
#1219= SPECIFIED_STATE('LOADSTATECORE_200','Core Loads Specified State'
);
#1220= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#325,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1221= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#1224= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#349,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-250.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1226= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#373,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-250.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
```

```
#1228= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#397,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-250.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1230= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#421,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1232= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_200',
'Item Factor State');
#1233= STATE_RELATIONSHIP('LOADSTATEITEM_200 is related to OverallComp',
'',#1217,#1232);
#1234= STATE_COMPONENT('ItemComp_21_200','',#1232,1.);
#1236= STATE_RELATIONSHIP(
'LOADSTATECORE_200 is related to ItemComp_21_200','',#1234,#1219);
#1237= STATE_RELATIONSHIP(
'LOADSTATECOMBINATION_21 is related to Step 1 Base Specified State','',
#745,#1215);
#1238= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 1','',
#745);
#1239= OUTPUT_REQUEST_STATE('Step 1 Output Request State','',(#744));
#1241= NODAL_FREEDOM_VALUES(#1239,#1245,#35,#1242,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#1242= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(
.Y_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTATION.)));
#1245= NODE_GROUP('','ALL',#67,(#80,#89,#93,#97,#101,#105,#109,#113,#117
,#121,#125,#129,#133,#137,#141,#145,#149,#153,#157,#161,#165,#169,#173,
#177,#181,#185,#189,#193,#197,#201,#205,#209,#213,#217,#221,#225,#229,
#233,#237,#241,#245,#249,#253,#257,#261,#265,#269,#273,#277,#281,#285,
#289,#293,#297,#301,#305,#309,#313,#317,#321,#325,#329,#333,#337,#341,
#345,#349,#353,#357,#361,#365,#369,#373,#377,#381,#385,#389,#393,#397,
#401,#405,#409,#413,#417,#421));
#1246= ID_ATTRIBUTE('ALL',#1245);
#1248= SURFACE_3D_ELEMENT_LOCATION_POINT_VOLUME_VARIABLE_VALUES(#1239,
#468,.F.,(#1249,#1262,#1267,#1272,#1277,#1282,#1286,#1288,#1290,#1292),
VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#1249= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1250,$);
#1250= SURFACE_VOLUME_ELEMENT_LOCATION(#1251,#1259);
#1251= SURFACE_ELEMENT_LOCATION(#1252);
#1252= FEA_PARAMETRIC_POINT('sQuadN_0',(0.,0.,0.));
#1254= REPRESENTATION('dpr0',(#1252,#1265,#1270,#1275,#1280,#1298),#1255
);
#1255= PARAMETRIC_REPRESENTATION_CONTEXT('FEA parametric context',
'Parametric');
#1259= SURFACE_SECTION_ELEMENT_LOCATION_DIMENSIONLESS(.F.,(-1.));
#1262= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1263,$);
#1263= SURFACE_VOLUME_ELEMENT_LOCATION(#1264,#1259);
#1264= SURFACE_ELEMENT_LOCATION(#1265);
#1265= FEA_PARAMETRIC_POINT('sQuadN_1',(-1.,-1.,0.));
#1267= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1268,$);
#1268= SURFACE_VOLUME_ELEMENT_LOCATION(#1269,#1259);
#1269= SURFACE_ELEMENT_LOCATION(#1270);
#1270= FEA_PARAMETRIC_POINT('sQuadN_2',(1.,-1.,0.));
#1272= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1273,$);
#1273= SURFACE_VOLUME_ELEMENT_LOCATION(#1274,#1259);
#1274= SURFACE_ELEMENT_LOCATION(#1275);
#1275= FEA_PARAMETRIC_POINT('sQuadN_3',(1.,1.,0.));
#1277= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1278,$);
#1278= SURFACE_VOLUME_ELEMENT_LOCATION(#1279,#1259);
#1279= SURFACE_ELEMENT_LOCATION(#1280);
#1280= FEA_PARAMETRIC_POINT('sQuadN_4',(-1.,1.,0.));
#1282= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1283,$);
```

```
#1283= SURFACE_VOLUME_ELEMENT_LOCATION(#1251,#1284);
#1284= SURFACE_SECTION_ELEMENT_LOCATION_DIMENSIONLESS(.F.,(1.));
#1286= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1287,$);
#1287= SURFACE_VOLUME_ELEMENT_LOCATION(#1264,#1284);
#1288= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1289,$);
#1289= SURFACE_VOLUME_ELEMENT_LOCATION(#1269,#1284);
#1290= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1291,$);
#1291= SURFACE_VOLUME_ELEMENT_LOCATION(#1274,#1284);
#1292= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1293,$);
#1293= SURFACE_VOLUME_ELEMENT_LOCATION(#1279,#1284);
#1294= SURFACE_3D_ELEMENT_LOCATION_POINT_VOLUME_VARIABLE_VALUES(#1239,
#594,.F.,(#1295,#1303),VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#1295= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1296,$);
#1296= SURFACE_VOLUME_ELEMENT_LOCATION(#1297,#1300);
#1297= SURFACE_ELEMENT_LOCATION(#1298);
#1298= FEA_PARAMETRIC_POINT('sTriN_0',(0.,0.,0.));
#1300= SURFACE_SECTION_ELEMENT_LOCATION_DIMENSIONLESS(.F.,(-1.));
#1303= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1304,$);
#1304= SURFACE_VOLUME_ELEMENT_LOCATION(#1297,#1305);
#1305= SURFACE_SECTION_ELEMENT_LOCATION_DIMENSIONLESS(.F.,(1.));
#1307= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#740,'STATIC STEP 2',2,#743,
'',#1797);
#1308= SPECIFIED_STATE('Step 2 Base Specified State',
'Relating Specified State');
#1309= SPECIFIED_STATE('SPCCASE_12_22_2','Aggregator SPC 12 Step 2');
#1310= SPECIFIED_STATE('SPCVALSTATE_100_22','');
#1311= STATE_RELATIONSHIP(
'SPCVALSTATE_100_22 is related to SPCCASE_12_22_2','',#1309,#1310);
#1312= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1310,#1313,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1313= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.90',(#1307),#173,#35,(
#754,#757,#758),'');
#1317= SPECIFIED_STATE('SPCVALSTATE_102_22','');
#1318= STATE_RELATIONSHIP(
'SPCVALSTATE_102_22 is related to SPCCASE_12_22_2','',#1309,#1317);
#1319= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1317,#1320,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1320= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.91',(#1307),#80,#35,(
#754,#757,#758),'');
#1324= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1317,#1325,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1325= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.92',(#1307),#129,#35,(
#754,#757,#758),'');
#1329= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1317,#1330,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1330= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.93',(#1307),#217,#35,(
#754,#757,#758),'');
#1334= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1317,#1335,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1335= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.94',(#1307),#261,#35,(
#754,#757,#758),'');
#1339= SPECIFIED_STATE('SPCVALSTATE_110_22','');
#1340= STATE_RELATIONSHIP(
'SPCVALSTATE_110_22 is related to SPCCASE_12_22_2','',#1309,#1339);
#1341= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1342,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1342= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.95',(#1307),#80,#35,(
#790,#793),'');
#1346= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1347,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
```

```
#1347= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.96',(#1307),#89,#35,(
#790,#793),'');
#1351= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1352,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1352= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.97',(#1307),#93,#35,(
#790,#793),'');
#1356= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1357,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1357= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.98',(#1307),#97,#35,(
#790,#793),'');
#1361= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1362,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1362= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.99',(#1307),#101,#35,(
#790,#793),'');
#1366= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1367,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1367= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.100',(#1307),#105,#35,(
#790,#793),'');
#1371= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1372,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1372= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.101',(#1307),#109,#35,(
#790,#793),'');
#1376= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1377,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1377= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.102',(#1307),#113,#35,(
#790,#793),'');
#1381= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1382,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1382= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.103',(#1307),#117,#35,(
#790,#793),'');
#1386= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1387,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1387= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.104',(#1307),#121,#35,(
#790,#793),'');
#1391= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1392,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1392= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.105',(#1307),#125,#35,(
#790,#793),'');
#1396= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1397,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1397= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.106',(#1307),#129,#35,(
#790,#793),'');
#1401= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1402,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1402= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.107',(#1307),#133,#35,(
#790,#793),'');
#1406= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1407,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1407= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.108',(#1307),#137,#35,(
#790,#793),'');
#1411= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1412,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1412= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.109',(#1307),#141,#35,(
#790,#793),'');
#1416= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1417,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1417= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.110',(#1307),#145,#35,(
#790,#793),'');
#1421= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1422,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1422= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.111',(#1307),#149,#35,(
#790,#793),'');
#1426= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1427,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1427= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.112',(#1307),#153,#35,(
#790,#793),'');
#1431= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1432,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1432= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.113',(#1307),#157,#35,(
#790,#793),'');
#1436= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1437,#786,(
```

```
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1437= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.114',(#1307),#161,#35,(
#790,#793),'');
#1441= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1442,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1442= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.115',(#1307),#165,#35,(
#790,#793),'');
#1446= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1447,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1447= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.116',(#1307),#169,#35,(
#790,#793),'');
#1451= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1452,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1452= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.117',(#1307),#173,#35,(
#790,#793),'');
#1456= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1457,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1457= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.118',(#1307),#177,#35,(
#790,#793),'');
#1461= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1462,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1462= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.119',(#1307),#181,#35,(
#790,#793),'');
#1466= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1467,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1467= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.120',(#1307),#185,#35,(
#790,#793),'');
#1471= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1472,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1472= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.121',(#1307),#189,#35,(
#790,#793),'');
#1476= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1477,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1477= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.122',(#1307),#193,#35,(
#790,#793),'');
#1481= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1482,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1482= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.123',(#1307),#197,#35,(
#790,#793),'');
#1486= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1487,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1487= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.124',(#1307),#201,#35,(
#790,#793),'');
#1491= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1492,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1492= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.125',(#1307),#205,#35,(
#790,#793),'');
#1496= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1497,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1497= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.126',(#1307),#209,#35,(
#790,#793),'');
#1501= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1502,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1502= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.127',(#1307),#213,#35,(
#790,#793),'');
#1506= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1507,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1507= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.128',(#1307),#217,#35,(
#790,#793),'');
#1511= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1512,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1512= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.129',(#1307),#221,#35,(
#790,#793),'');
#1516= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1517,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1517= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.130',(#1307),#225,#35,(
#790,#793),'');
#1521= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1522,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1522= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.131',(#1307),#229,#35,(
#790,#793),'');
```

```
#1526= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1527,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1527= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.132',(#1307),#233,#35,(
#790,#793),'');
#1531= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1532,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1532= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.133',(#1307),#237,#35,(
#790,#793),'');
#1536= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1537,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1537= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.134',(#1307),#241,#35,(
#790,#793),'');
#1541= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1542,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1542= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.135',(#1307),#245,#35,(
#790,#793),'');
#1546= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1547,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1547= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.136',(#1307),#249,#35,(
#790,#793),'');
#1551= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1552,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1552= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.137',(#1307),#253,#35,(
#790,#793),'');
#1556= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1557,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1557= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.138',(#1307),#257,#35,(
#790,#793),'');
#1561= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1562,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1562= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.139',(#1307),#261,#35,(
#790,#793),'');
#1566= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1567,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1567= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.140',(#1307),#265,#35,(
#790,#793),'');
#1571= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1572,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1572= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.141',(#1307),#269,#35,(
#790,#793),'');
#1576= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1577,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1577= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.142',(#1307),#273,#35,(
#790,#793),'');
#1581= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1582,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1582= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.143',(#1307),#277,#35,(
#790,#793),'');
#1586= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1587,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1587= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.144',(#1307),#281,#35,(
#790,#793),'');
#1591= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1592,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1592= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.145',(#1307),#285,#35,(
#790,#793),'');
#1596= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1597,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1597= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.146',(#1307),#289,#35,(
#790,#793),'');
#1601= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1602,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1602= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.147',(#1307),#293,#35,(
#790,#793),'');
#1606= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1607,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1607= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.148',(#1307),#297,#35,(
#790,#793),'');
#1611= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1612,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1612= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.149',(#1307),#301,#35,(
```

```
#790,#793),'');
#1616= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1617,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1617= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.150',(#1307),#305,#35,(
#790,#793),'');
#1621= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1622,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1622= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.151',(#1307),#309,#35,(
#790,#793),'');
#1626= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1627,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1627= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.152',(#1307),#313,#35,(
#790,#793),'');
#1631= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1632,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1632= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.153',(#1307),#317,#35,(
#790,#793),'');
#1636= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1637,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1637= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.154',(#1307),#321,#35,(
#790,#793),'');
#1641= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1642,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1642= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.155',(#1307),#325,#35,(
#790,#793),'');
#1646= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1647,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1647= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.156',(#1307),#329,#35,(
#790,#793),'');
#1651= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1652,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1652= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.157',(#1307),#333,#35,(
#790,#793),'');
#1656= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1657,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1657= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.158',(#1307),#337,#35,(
#790,#793),'');
#1661= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1662,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1662= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.159',(#1307),#341,#35,(
#790,#793),'');
#1666= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1667,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1667= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.160',(#1307),#345,#35,(
#790,#793),'');
#1671= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1672,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1672= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.161',(#1307),#349,#35,(
#790,#793),'');
#1676= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1677,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1677= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.162',(#1307),#353,#35,(
#790,#793),'');
#1681= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1682,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1682= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.163',(#1307),#357,#35,(
#790,#793),'');
#1686= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1687,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1687= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.164',(#1307),#361,#35,(
#790,#793),'');
#1691= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1692,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1692= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.165',(#1307),#365,#35,(
#790,#793),'');
#1696= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1697,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1697= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.166',(#1307),#369,#35,(
#790,#793),'');
#1701= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1702,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
```

```
#1702= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.167',(#1307),#373,#35,(
#790,#793),'');
#1706= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1707,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1707= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.168',(#1307),#377,#35,(
#790,#793),'');
#1711= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1712,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1712= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.169',(#1307),#381,#35,(
#790,#793),'');
#1716= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1717,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1717= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.170',(#1307),#385,#35,(
#790,#793),'');
#1721= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1722,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1722= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.171',(#1307),#389,#35,(
#790,#793),'');
#1726= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1727,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1727= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.172',(#1307),#393,#35,(
#790,#793),'');
#1731= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1732,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1732= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.173',(#1307),#397,#35,(
#790,#793),'');
#1736= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1737,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1737= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.174',(#1307),#401,#35,(
#790,#793),'');
#1741= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1742,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1742= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.175',(#1307),#405,#35,(
#790,#793),'');
#1746= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1747,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1747= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.176',(#1307),#409,#35,(
#790,#793),'');
#1751= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1752,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1752= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.177',(#1307),#413,#35,(
#790,#793),'');
#1756= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1757,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1757= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.178',(#1307),#417,#35,(
#790,#793),'');
#1761= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1339,#1762,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1762= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.179',(#1307),#421,#35,(
#790,#793),'');
#1766= STATE_RELATIONSHIP(
'SPCCASE_12_22_2 is related to Step 2 Base Specified State','',#1308,
#1309);
#1767= LINEARLY_SUPERIMPOSED_STATE('LOADSTATECOMBINATION_22',
'Overall Factor Combined State');
#1768= STATE_COMPONENT('OverallComp','',#1767,1.);
#1770= SPECIFIED_STATE('LOADSTATECORE_300','Core Loads Specified State'
);
#1771= NODAL_FREEDOM_ACTION_DEFINITION(#1770,#301,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1773= NODAL_FREEDOM_ACTION_DEFINITION(#1770,#421,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1775= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_300',
'Item Factor State');
#1776= STATE_RELATIONSHIP('LOADSTATEITEM_300 is related to OverallComp',
'',#1768,#1775);
#1777= STATE_COMPONENT('ItemComp_22_300','',#1775,1.);
#1779= STATE_RELATIONSHIP(
'LOADSTATECORE_300 is related to ItemComp_22_300','',#1777,#1770);
```

```
#1780= SPECIFIED_STATE('LOADSTATECORE_400','Core Loads Specified State'
);
#1781= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#401,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1783= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#405,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1785= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#409,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1787= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#413,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1789= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#417,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#1791= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_400',
'Item Factor State');
#1792= STATE_RELATIONSHIP('LOADSTATEITEM_400 is related to OverallComp',
'',#1768,#1791);
#1793= STATE_COMPONENT('ItemComp_22_400','',#1791,1.);
#1795= STATE_RELATIONSHIP(
'LOADSTATECORE_400 is related to ItemComp_22_400','',#1793,#1780);
#1796= STATE_RELATIONSHIP(
'LOADSTATECOMBINATION_22 is related to Step 2 Base Specified State','',
#1308,#1767);
#1797= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 2','',
#1308);
#1798= OUTPUT_REQUEST_STATE('Step 2 Output Request State','',(#1307));
#1800= NODAL_FREEDOM_VALUES(#1798,#1245,#35,#1242,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#1802= SURFACE_3D_ELEMENT_LOCATION_POINT_VOLUME_VARIABLE_VALUES(#1798,
#468,.F.,(#1803,#1805,#1806,#1807,#1808,#1809,#1810,#1811,#1812,#1813),
VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#1803= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1250,$);
#1805= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1263,$);
#1806= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1268,$);
#1807= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1273,$);
#1808= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1278,$);
#1809= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1283,$);
#1810= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1287,$);
#1811= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1289,$);
#1812= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1291,$);
#1813= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1293,$);
#1814= SURFACE_3D_ELEMENT_LOCATION_POINT_VOLUME_VARIABLE_VALUES(#1798,
#594,.F.,(#1815,#1817),VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#1815= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1296,$);
#1817= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1304,$);
#1818= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#740,'STATIC STEP 3',3,#743,
'',#2291);
#1819= SPECIFIED_STATE('Step 3 Base Specified State',
'Relating Specified State');
#1820= SPECIFIED_STATE('SPCCASE_12_23_3','Aggregator SPC 12 Step 3');
#1821= SPECIFIED_STATE('SPCVALSTATE_100_23','');
#1822= STATE_RELATIONSHIP(
'SPCVALSTATE_100_23 is related to SPCCASE_12_23_3','',#1820,#1821);
```

```
#1823= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1821,#1824,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1824= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.180',(#1818),#173,#35,(
#754,#757,#758),'');
#1828= SPECIFIED_STATE('SPCVALSTATE_102_23','');
#1829= STATE_RELATIONSHIP(
'SPCVALSTATE_102_23 is related to SPCCASE_12_23_3','',#1820,#1828);
#1830= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1828,#1831,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1831= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.181',(#1818),#80,#35,(
#754,#757,#758),'');
#1835= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1828,#1836,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1836= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.182',(#1818),#129,#35,(
#754,#757,#758),'');
#1840= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1828,#1841,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1841= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.183',(#1818),#217,#35,(
#754,#757,#758),'');
#1845= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1828,#1846,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#1846= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.184',(#1818),#261,#35,(
#754,#757,#758),'');
#1850= SPECIFIED_STATE('SPCVALSTATE_110_23','');
#1851= STATE_RELATIONSHIP(
'SPCVALSTATE_110_23 is related to SPCCASE_12_23_3','',#1820,#1850);
#1852= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1853,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1853= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.185',(#1818),#80,#35,(
#790,#793),'');
#1857= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1858,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1858= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.186',(#1818),#89,#35,(
#790,#793),'');
#1862= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1863,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1863= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.187',(#1818),#93,#35,(
#790,#793),'');
#1867= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1868,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1868= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.188',(#1818),#97,#35,(
#790,#793),'');
#1872= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1873,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1873= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.189',(#1818),#101,#35,(
#790,#793),'');
#1877= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1878,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1878= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.190',(#1818),#105,#35,(
#790,#793),'');
#1882= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1883,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1883= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.191',(#1818),#109,#35,(
#790,#793),'');
#1887= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1888,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1888= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.192',(#1818),#113,#35,(
#790,#793),'');
#1892= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1893,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1893= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.193',(#1818),#117,#35,(
#790,#793),'');
#1897= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1898,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1898= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.194',(#1818),#121,#35,(
#790,#793),'');
```

```
#1902= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1903,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1903= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.195',(#1818),#125,#35,(
#790,#793),'');
#1907= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1908,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1908= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.196',(#1818),#129,#35,(
#790,#793),'');
#1912= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1913,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1913= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.197',(#1818),#133,#35,(
#790,#793),'');
#1917= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1918,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1918= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.198',(#1818),#137,#35,(
#790,#793),'');
#1922= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1923,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1923= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.199',(#1818),#141,#35,(
#790,#793),'');
#1927= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1928,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1928= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.200',(#1818),#145,#35,(
#790,#793),'');
#1932= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1933,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1933= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.201',(#1818),#149,#35,(
#790,#793),'');
#1937= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1938,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1938= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.202',(#1818),#153,#35,(
#790,#793),'');
#1942= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1943,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1943= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.203',(#1818),#157,#35,(
#790,#793),'');
#1947= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1948,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1948= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.204',(#1818),#161,#35,(
#790,#793),'');
#1952= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1953,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1953= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.205',(#1818),#165,#35,(
#790,#793),'');
#1957= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1958,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1958= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.206',(#1818),#169,#35,(
#790,#793),'');
#1962= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1963,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1963= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.207',(#1818),#173,#35,(
#790,#793),'');
#1967= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1968,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1968= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.208',(#1818),#177,#35,(
#790,#793),'');
#1972= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1973,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1973= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.209',(#1818),#181,#35,(
#790,#793),'');
#1977= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1978,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1978= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.210',(#1818),#185,#35,(
#790,#793),'');
#1982= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1983,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1983= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.211',(#1818),#189,#35,(
#790,#793),'');
#1987= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1988,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1988= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.212',(#1818),#193,#35,(
```

```
#790,#793),'');
#1992= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1993,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1993= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.213',(#1818),#197,#35,(
#790,#793),'');
#1997= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#1998,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#1998= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.214',(#1818),#201,#35,(
#790,#793),'');
#2002= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2003,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2003= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.215',(#1818),#205,#35,(
#790,#793),'');
#2007= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2008,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2008= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.216',(#1818),#209,#35,(
#790,#793),'');
#2012= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2013,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2013= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.217',(#1818),#213,#35,(
#790,#793),'');
#2017= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2018,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2018= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.218',(#1818),#217,#35,(
#790,#793),'');
#2022= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2023,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2023= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.219',(#1818),#221,#35,(
#790,#793),'');
#2027= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2028,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2028= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.220',(#1818),#225,#35,(
#790,#793),'');
#2032= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2033,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2033= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.221',(#1818),#229,#35,(
#790,#793),'');
#2037= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2038,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2038= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.222',(#1818),#233,#35,(
#790,#793),'');
#2042= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2043,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2043= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.223',(#1818),#237,#35,(
#790,#793),'');
#2047= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2048,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2048= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.224',(#1818),#241,#35,(
#790,#793),'');
#2052= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2053,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2053= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.225',(#1818),#245,#35,(
#790,#793),'');
#2057= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2058,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2058= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.226',(#1818),#249,#35,(
#790,#793),'');
#2062= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2063,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2063= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.227',(#1818),#253,#35,(
#790,#793),'');
#2067= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2068,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2068= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.228',(#1818),#257,#35,(
#790,#793),'');
#2072= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2073,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2073= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.229',(#1818),#261,#35,(
#790,#793),'');
#2077= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2078,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
```

```
#2078= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.230',(#1818),#265,#35,(
#790,#793),'');
#2082= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2083,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2083= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.231',(#1818),#269,#35,(
#790,#793),'');
#2087= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2088,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2088= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.232',(#1818),#273,#35,(
#790,#793),'');
#2092= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2093,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2093= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.233',(#1818),#277,#35,(
#790,#793),'');
#2097= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2098,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2098= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.234',(#1818),#281,#35,(
#790,#793),'');
#2102= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2103,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2103= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.235',(#1818),#285,#35,(
#790,#793),'');
#2107= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2108,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2108= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.236',(#1818),#289,#35,(
#790,#793),'');
#2112= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2113,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2113= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.237',(#1818),#293,#35,(
#790,#793),'');
#2117= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2118,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2118= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.238',(#1818),#297,#35,(
#790,#793),'');
#2122= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2123,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2123= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.239',(#1818),#301,#35,(
#790,#793),'');
#2127= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2128,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2128= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.240',(#1818),#305,#35,(
#790,#793),'');
#2132= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2133,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2133= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.241',(#1818),#309,#35,(
#790,#793),'');
#2137= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2138,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2138= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.242',(#1818),#313,#35,(
#790,#793),'');
#2142= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2143,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2143= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.243',(#1818),#317,#35,(
#790,#793),'');
#2147= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2148,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2148= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.244',(#1818),#321,#35,(
#790,#793),'');
#2152= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2153,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2153= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.245',(#1818),#325,#35,(
#790,#793),'');
#2157= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2158,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2158= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.246',(#1818),#329,#35,(
#790,#793),'');
#2162= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2163,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2163= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.247',(#1818),#333,#35,(
#790,#793),'');
#2167= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2168,#786,(
```

```
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2168= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.248',(#1818),#337,#35,(
#790,#793),'');
#2172= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2173,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2173= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.249',(#1818),#341,#35,(
#790,#793),'');
#2177= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2178,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2178= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.250',(#1818),#345,#35,(
#790,#793),'');
#2182= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2183,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2183= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.251',(#1818),#349,#35,(
#790,#793),'');
#2187= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2188,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2188= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.252',(#1818),#353,#35,(
#790,#793),'');
#2192= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2193,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2193= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.253',(#1818),#357,#35,(
#790,#793),'');
#2197= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2198,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2198= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.254',(#1818),#361,#35,(
#790,#793),'');
#2202= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2203,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2203= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.255',(#1818),#365,#35,(
#790,#793),'');
#2207= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2208,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2208= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.256',(#1818),#369,#35,(
#790,#793),'');
#2212= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2213,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2213= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.257',(#1818),#373,#35,(
#790,#793),'');
#2217= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2218,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2218= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.258',(#1818),#377,#35,(
#790,#793),'');
#2222= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2223,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2223= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.259',(#1818),#381,#35,(
#790,#793),'');
#2227= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2228,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2228= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.260',(#1818),#385,#35,(
#790,#793),'');
#2232= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2233,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2233= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.261',(#1818),#389,#35,(
#790,#793),'');
#2237= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2238,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2238= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.262',(#1818),#393,#35,(
#790,#793),'');
#2242= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2243,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2243= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.263',(#1818),#397,#35,(
#790,#793),'');
#2247= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2248,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2248= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.264',(#1818),#401,#35,(
#790,#793),'');
#2252= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2253,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2253= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.265',(#1818),#405,#35,(
#790,#793),'');
```

```
#2257= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2258,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2258= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.266',(#1818),#409,#35,(
#790,#793),'');
#2262= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2263,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2263= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.267',(#1818),#413,#35,(
#790,#793),'');
#2267= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2268,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2268= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.268',(#1818),#417,#35,(
#790,#793),'');
#2272= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#1850,#2273,#786,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2273= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_110.269',(#1818),#421,#35,(
#790,#793),'');
#2277= STATE_RELATIONSHIP(
'SPCCASE_12_23_3 is related to Step 3 Base Specified State','',#1819,
#1820);
#2278= LINEARLY_SUPERIMPOSED_STATE('LOADSTATECOMBINATION_23',
'Overall Factor Combined State');
#2279= STATE_COMPONENT('OverallComp','',#2278,1.);
#2281= STATE_RELATIONSHIP('LOADSTATEITEM_200 is related to OverallComp',
'',#2279,#1232);
#2282= STATE_COMPONENT('ItemComp_23_200','',#1232,1.);
#2283= STATE_RELATIONSHIP(
'LOADSTATECORE_200 is related to ItemComp_23_200','',#2282,#1219);
#2284= STATE_RELATIONSHIP('LOADSTATEITEM_300 is related to OverallComp',
'',#2279,#1775);
#2285= STATE_COMPONENT('ItemComp_23_300','',#1775,1.);
#2286= STATE_RELATIONSHIP(
'LOADSTATECORE_300 is related to ItemComp_23_300','',#2285,#1770);
#2287= STATE_RELATIONSHIP('LOADSTATEITEM_400 is related to OverallComp',
'',#2279,#1791);
#2288= STATE_COMPONENT('ItemComp_23_400','',#1791,1.);
#2289= STATE_RELATIONSHIP(
'LOADSTATECORE_400 is related to ItemComp_23_400','',#2288,#1780);
#2290= STATE_RELATIONSHIP(
'LOADSTATECOMBINATION_23 is related to Step 3 Base Specified State','',
#1819,#2278);
#2291= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 3','',
#1819);
#2292= OUTPUT_REQUEST_STATE('Step 3 Output Request State','',(#1818));
#2294= NODAL_FREEDOM_VALUES(#2292,#1245,#35,#1242,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#2296= SURFACE_3D_ELEMENT_LOCATION_POINT_VOLUME_VARIABLE_VALUES(#2292,
#468,.F.,(#2297,#2299,#2300,#2301,#2302,#2303,#2304,#2305,#2306,#2307),
VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2297= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1250,$);
#2299= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1263,$);
#2300= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1268,$);
#2301= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1273,$);
#2302= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1278,$);
#2303= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1283,$);
#2304= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1287,$);
#2305= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1289,$);
#2306= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1291,$);
#2307= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1293,$);
#2308= SURFACE_3D_ELEMENT_LOCATION_POINT_VOLUME_VARIABLE_VALUES(#2292,
```

```
#594,.F.,(#2309,#2311),VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2309= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1296,$);
#2311= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1304,$);
#2312= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#740,'STATIC STEP 4',4,#743,
'',#2382);
#2313= SPECIFIED_STATE('Step 4 Base Specified State',
'Relating Specified State');
#2314= SPECIFIED_STATE('SPCCASE_103_500_4','Aggregator SPC 103 Step 4');
#2315= SPECIFIED_STATE('SPCVALSTATE_103_500','');
#2316= STATE_RELATIONSHIP(
'SPCVALSTATE_103_500 is related to SPCCASE_103_500_4','',#2314,#2315);
#2317= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2318,#750,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2318= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.270',(#2312),#173,#35,(
#754,#757,#758),'');
#2322= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2325,#2323,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.)));
#2323= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#2325= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.271',(#2312),#373,#35,(
#757,#758),'');
#2329= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2332,#2330,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2330= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#2332= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.272',(#2312),#80,#35,(
#758),'');
#2336= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2337,#2330,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2337= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.273',(#2312),#129,#35,(
#758),'');
#2341= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2342,#2330,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2342= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.274',(#2312),#217,#35,(
#758),'');
#2346= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2347,#2330,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2347= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.275',(#2312),#261,#35,(
#758),'');
#2351= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2352,#2330,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2352= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.276',(#2312),#325,#35,(
#758),'');
#2356= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2357,#2330,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2357= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.277',(#2312),#349,#35,(
#758),'');
#2361= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2362,#2330,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2362= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.278',(#2312),#397,#35,(
#758),'');
#2366= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2315,#2367,#2330,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2367= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_103.279',(#2312),#421,#35,(
#758),'');
#2371= STATE_RELATIONSHIP(
'SPCCASE_103_500_4 is related to Step 4 Base Specified State','',#2313,
#2314);
#2372= SPECIFIED_STATE('LOADSTATECORE_500','Core Loads Specified State'
);
#2373=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#491,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2374=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#494,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2375=
```

```
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#521,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2376=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#524,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2377=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#551,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2378=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#554,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2379=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#581,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2380=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#584,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2381= STATE_RELATIONSHIP(
'LOADSTATECORE_500 is related to Step 4 Base Specified State','',#2313,
#2372);
#2382= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 4','',
#2313);
#2383= OUTPUT_REQUEST_STATE('Step 4 Output Request State','',(#2312));
#2385= NODAL_FREEDOM_VALUES(#2383,#1245,#35,#1242,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#2387= SURFACE_3D_ELEMENT_LOCATION_POINT_VOLUME_VARIABLE_VALUES(#2383,
#468,.F.,(#2388,#2390,#2391,#2392,#2393,#2394,#2395,#2396,#2397,#2398),
VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2388= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1250,$);
#2390= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1263,$);
#2391= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1268,$);
#2392= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1273,$);
#2393= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1278,$);
#2394= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1283,$);
#2395= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1287,$);
#2396= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1289,$);
#2397= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1291,$);
#2398= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1293,$);
#2399= SURFACE_3D_ELEMENT_LOCATION_POINT_VOLUME_VARIABLE_VALUES(#2383,
#594,.F.,(#2400,#2402),VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2400= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1296,$);
#2402= SURFACE_3D_ELEMENT_VALUE_AND_VOLUME_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#1304,$);
#2403= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#325,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2405= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#349,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-250.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2407= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#373,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-250.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
```

```
#2409= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#397,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-250.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2411= NODAL_FREEDOM_ACTION_DEFINITION(#1219,#421,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2413= NODAL_FREEDOM_ACTION_DEFINITION(#1770,#301,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2415= NODAL_FREEDOM_ACTION_DEFINITION(#1770,#421,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2417= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#401,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2419= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#405,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2421= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#409,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2423= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#413,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2425= NODAL_FREEDOM_ACTION_DEFINITION(#1780,#417,#35,#1221,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-20.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2427=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#491,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2428=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#494,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2429=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#521,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2430=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#524,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2431=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#551,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2432=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#554,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2433=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#581,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
#2434=
SURFACE_3D_ELEMENT_BOUNDARY_CONSTANT_SPECIFIED_SURFACE_VARIABLE_VALUE(
#2372,#584,CONTEXT_DEPENDENT_MEASURE(125.),
BOUNDARY_SURFACE_SCALAR_VARIABLE(.PRESSURE.),2,$);
ENDSEC;

END-ISO-10303-21;
```

11.4 ATS4m4.stp

```
ISO-10303-21;
HEADER;

/*****************************************************************************************
* Creation date:                  Tue Feb 07 13:53:33 2017
* Schema:                         AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF
*****************************************************************************************/

FILE_DESCRIPTION((''),'2;1');
FILE_NAME(
'O:\\projects\\Crystal\\software_FEM\\converters\\examples\\db\\e.DataRepository.Fem_Model'

,
          '2017-02-07T13:53:33',
          ('ANONYMOUS USER'),
          ('ANONYMOUS ORGANISATION'),
          'EDMsix Version 2.0100.a14 Jan 31 2017',
          '',
          '');
FILE_SCHEMA(('AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF'));
ENDSEC;

DATA;
#1= APPLICATION_CONTEXT(
'AP209_MULTIDISCIPLINARY_ANALYSIS_AND_DESIGN_MIM_LF');
#3= ID_ATTRIBUTE('to be decided',#1);
#4= DESCRIPTION_ATTRIBUTE('to be decided',#1);
#5= APPLICATION_PROTOCOL_DEFINITION('international standard',
'ap209_multidisciplinary_analysis_and_design',2011,#1);
#6= DIMENSIONAL_EXPONENTS(0.,0.,0.,0.,0.,0.,0.);
#7= /* LENGTH_UNIT+SI_UNIT */(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT($,
.METRE.));
#8= /* MASS_UNIT+SI_UNIT */(MASS_UNIT()NAMED_UNIT(*)SI_UNIT(.KILO.,
.GRAM.));
#9= /* SI_UNIT+TIME_UNIT */(NAMED_UNIT(*)SI_UNIT($,.SECOND.)TIME_UNIT()
);
#10= /* SI_UNIT+THERMODYNAMIC_TEMPERATURE_UNIT */(NAMED_UNIT(*)SI_UNIT($
,.DEGREE_CELSIUS.)THERMODYNAMIC_TEMPERATURE_UNIT());
#11= /* PLANE_ANGLE_UNIT+SI_UNIT */(NAMED_UNIT(*)PLANE_ANGLE_UNIT()
SI_UNIT($,.RADIAN.));
#12= SI_FORCE_UNIT((#13,#14,#15),*,$,.NEWTON.);
#13= DERIVED_UNIT_ELEMENT(#8,1.);
#14= DERIVED_UNIT_ELEMENT(#7,1.);
#15= DERIVED_UNIT_ELEMENT(#9,-2.);
#17= SI_FREQUENCY_UNIT((#18),*,$,.HERTZ.);
#18= DERIVED_UNIT_ELEMENT(#9,-1.);
#20= SI_PRESSURE_UNIT((#13,#21,#15),*,$,.PASCAL.);
#21= DERIVED_UNIT_ELEMENT(#7,-1.);
#23= SI_ENERGY_UNIT((#13,#24,#15),*,$,.JOULE.);
#24= DERIVED_UNIT_ELEMENT(#7,2.);
#26= SI_POWER_UNIT((#13,#24,#27),*,$,.WATT.);
#27= DERIVED_UNIT_ELEMENT(#9,-3.);
#29= DIRECTION('RefX',(1.,0.,0.));
#31= DIRECTION('AxisZ',(0.,0.,1.));
#33= CARTESIAN_POINT('',(0.,0.,0.));
#35= FEA_AXIS2_PLACEMENT_3D('0',#33,#31,#29,.CARTESIAN.,
'FEA_BASIC_COORD_SYSTEM.0');
#36= /* GEOMETRIC_REPRESENTATION_CONTEXT+GLOBAL_UNIT_ASSIGNED_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((#23,
#12,#17,#7,#8,#11,#26,#20,#10,#9))REPRESENTATION_CONTEXT(
'FEA_BASIC_COORD_SYSTEM.0','3d'));
#39= ORGANIZATION($,'default-organization',$);
#40= PRODUCT('default-id.0','default-fea-part','',(#42));
#42= PRODUCT_CONTEXT('design_context',#1,'design_context');
#44= PRODUCT_RELATED_PRODUCT_CATEGORY('part',$,(#40));
#45= ID_ATTRIBUTE('',#44);
#47= APPLIED_IDENTIFICATION_ASSIGNMENT('default-id.0',#48,(#50,#56,#40)
);
#48= IDENTIFICATION_ROLE('default-role',$);
```

```
#49= PRODUCT_RELATED_PRODUCT_CATEGORY('product',$,(#40));
#50= PRODUCT_DEFINITION_FORMATION('default-id.0','',#40);
#51= GROUP('id',$);
#52= APPLIED_CLASSIFICATION_ASSIGNMENT(#51,#54,(#47));
#54= CLASSIFICATION_ROLE('',$);
#56= PRODUCT_DEFINITION('default-id.0',$,#50,#57);
#57= PRODUCT_DEFINITION_CONTEXT('assembly definition',#1,
'life cycle stage');
#58= APPLIED_CLASSIFICATION_ASSIGNMENT(#59,#60,(#56));
#59= GROUP('default-gpvv-type',$);
#60= CLASSIFICATION_ROLE('view_type',$);
#63= ORGANIZATION_ROLE('id context');
#64= APPLIED_ORGANIZATION_ASSIGNMENT(#39,#63,(#47));
#66= NAME_ATTRIBUTE('',#56);
#67= FEA_MODEL_3D('Identification',(#35),#36,
'NASTRAN BDF Converter v0.0.0',('NASTRAN'),'AnalysisModelType');
#70= STRUCTURAL_RESPONSE_PROPERTY_DEFINITION_REPRESENTATION(#71,#67);
#71= STRUCTURAL_RESPONSE_PROPERTY('structural_response_property_id_1',$,
#72);
#72= FEA_MODEL_DEFINITION('fea_model_definition_id_1',$,#73,.U.);
#73= PRODUCT_DEFINITION_SHAPE('name',$,#56);
#74= CARTESIAN_POINT('0',(0.,0.,0.));
#76= DUMMY_NODE('0',(#74),#36,#67);
#78= CARTESIAN_POINT('1',(0.,-4.,-3.431520000000000E-8));
#80= NODE('1',(#78),#36,#67);
#81= SHAPE_DEFINITION_REPRESENTATION(#73,#82);
#82= POINT_REPRESENTATION('point_representation_name_1',(#83,#35),#36);
#83= NODE_SET('node_set_name_1',(#80,#89,#93,#97,#101,#105,#109,#113,
#117,#121,#125,#129,#133,#137,#141,#145,#149,#153,#157,#161,#165,#169,
#173,#177,#181,#185,#189,#193,#197,#201,#205,#209,#213,#217,#221,#225,
#229,#233,#237,#241,#245,#249,#253,#257,#261,#265,#269,#273,#277,#281,
#285,#289,#293,#297,#301,#305,#309,#313,#317,#321,#325,#329,#333,#337,
#341,#345,#349,#353,#357,#361,#365,#369,#373,#377,#381,#385,#389,#393,
#397,#401,#405,#409,#413,#417,#421,#425,#429,#433,#437,#441,#445,#449,
#453,#457,#461,#465,#469,#473,#477,#481,#485,#489,#493,#497,#501,#505,
#509,#513,#517,#521,#525,#529,#533,#537,#541,#545,#549,#553,#557,#561,
#565,#569,#573,#577,#581,#585,#589,#593,#597,#601,#605,#609,#613,#617,
#621,#625,#629,#633,#637,#641,#645,#649,#653,#657,#661,#665,#669,#673,
#677,#681,#685,#689,#693,#697,#701,#705,#709,#713,#717,#721,#725,#729,
#733,#737,#741,#745,#749,#753,#757,#761,#765,#769,#773,#777,#781,#785,
#789,#793,#797,#801,#805,#809,#813,#817,#821,#825,#829,#833,#837,#841,
#845,#849,#853,#857,#861,#865,#869,#873,#877,#881,#885,#889,#893,#897,
#901,#905,#909,#913,#917,#921,#925,#929,#933,#937,#941,#945,#949,#953,
#957,#961,#965,#969,#973,#977,#981,#985,#989,#993,#997,#1001,#1005,#1009
,#1013,#1017,#1021,#1025,#1029,#1033,#1037,#1041,#1045,#1049,#1053,#1057
,#1061,#1065,#1069,#1073,#1077,#1081,#1085,#1089,#1093,#1097,#1101));
#87= CARTESIAN_POINT('2',(0.,-3.,-3.431520000000000E-8));
#89= NODE('2',(#87),#36,#67);
#91= CARTESIAN_POINT('3',(0.,-2.,-3.431520000000000E-8));
#93= NODE('3',(#91),#36,#67);
#95= CARTESIAN_POINT('4',(0.,-1.,-3.431520000000000E-8));
#97= NODE('4',(#95),#36,#67);
#99= CARTESIAN_POINT('5',(0.,-6.013760000000000E-9,-3.431520000000000E-8
));
#101= NODE('5',(#99),#36,#67);
#103= CARTESIAN_POINT('6',(0.,-4.,1.));
#105= NODE('6',(#103),#36,#67);
#107= CARTESIAN_POINT('7',(0.,-3.,1.));
#109= NODE('7',(#107),#36,#67);
#111= CARTESIAN_POINT('8',(0.,-2.,1.));
#113= NODE('8',(#111),#36,#67);
#115= CARTESIAN_POINT('9',(0.,-1.,1.));
#117= NODE('9',(#115),#36,#67);
#119= CARTESIAN_POINT('10',(0.,-6.013760000000000E-9,1.));
#121= NODE('10',(#119),#36,#67);
#123= CARTESIAN_POINT('11',(0.,-4.,2.));
#125= NODE('11',(#123),#36,#67);
#127= CARTESIAN_POINT('12',(0.,-3.,2.));
#129= NODE('12',(#127),#36,#67);
#131= CARTESIAN_POINT('13',(0.,-2.,2.));
#133= NODE('13',(#131),#36,#67);
```

```
#135= CARTESIAN_POINT('14',(0.,-1.,2.));
#137= NODE('14',(#135),#36,#67);
#139= CARTESIAN_POINT('15',(0.,-6.013760000000000E-9,2.));
#141= NODE('15',(#139),#36,#67);
#143= CARTESIAN_POINT('16',(1.,-4.,-2.535450000000000E-8));
#145= NODE('16',(#143),#36,#67);
#147= CARTESIAN_POINT('17',(1.,-3.,-2.090090000000000E-8));
#149= NODE('17',(#147),#36,#67);
#151= CARTESIAN_POINT('18',(1.,-2.,-3.742290000000000E-8));
#153= NODE('18',(#151),#36,#67);
#155= CARTESIAN_POINT('19',(1.,-1.,-5.394500000000000E-8));
#157= NODE('19',(#155),#36,#67);
#159= CARTESIAN_POINT('20',(1.,-2.864290000000000E-9,
-4.949130000000000E-8));
#161= NODE('20',(#159),#36,#67);
#163= CARTESIAN_POINT('21',(1.,-4.,1.));
#165= NODE('21',(#163),#36,#67);
#167= CARTESIAN_POINT('22',(1.,-3.,1.));
#169= NODE('22',(#167),#36,#67);
#171= CARTESIAN_POINT('23',(1.,-2.,1.));
#173= NODE('23',(#171),#36,#67);
#175= CARTESIAN_POINT('24',(1.,-1.,1.));
#177= NODE('24',(#175),#36,#67);
#179= CARTESIAN_POINT('25',(1.,-3.284170000000000E-9,1.));
#181= NODE('25',(#179),#36,#67);
#183= CARTESIAN_POINT('26',(1.,-4.,2.));
#185= NODE('26',(#183),#36,#67);
#187= CARTESIAN_POINT('27',(1.,-3.,2.));
#189= NODE('27',(#187),#36,#67);
#191= CARTESIAN_POINT('28',(1.,-2.,2.));
#193= NODE('28',(#191),#36,#67);
#195= CARTESIAN_POINT('29',(1.,-1.,2.));
#197= NODE('29',(#195),#36,#67);
#199= CARTESIAN_POINT('30',(1.,-3.704040000000000E-9,2.));
#201= NODE('30',(#199),#36,#67);
#203= CARTESIAN_POINT('31',(2.,-4.,-1.639380000000000E-8));
#205= NODE('31',(#203),#36,#67);
#207= CARTESIAN_POINT('32',(2.,-3.,-7.486510000000000E-9));
#209= NODE('32',(#207),#36,#67);
#211= CARTESIAN_POINT('33',(2.,-2.,-4.053060000000000E-8));
#213= NODE('33',(#211),#36,#67);
#215= CARTESIAN_POINT('34',(2.,-1.,-7.357470000000001E-8));
#217= NODE('34',(#215),#36,#67);
#219= CARTESIAN_POINT('35',(2.,2.851770000000000E-10,
-6.466740000000000E-8));
#221= NODE('35',(#219),#36,#67);
#223= CARTESIAN_POINT('36',(2.,-4.,1.));
#225= NODE('36',(#223),#36,#67);
#227= CARTESIAN_POINT('37',(2.,-3.,1.));
#229= NODE('37',(#227),#36,#67);
#231= CARTESIAN_POINT('38',(2.,-2.,1.));
#233= NODE('38',(#231),#36,#67);
#235= CARTESIAN_POINT('39',(2.,-1.,1.));
#237= NODE('39',(#235),#36,#67);
#239= CARTESIAN_POINT('40',(2.,-5.545730000000000E-10,1.));
#241= NODE('40',(#239),#36,#67);
#243= CARTESIAN_POINT('41',(2.,-4.,2.));
#245= NODE('41',(#243),#36,#67);
#247= CARTESIAN_POINT('42',(2.,-3.,2.));
#249= NODE('42',(#247),#36,#67);
#251= CARTESIAN_POINT('43',(2.,-2.,2.));
#253= NODE('43',(#251),#36,#67);
#255= CARTESIAN_POINT('44',(2.,-1.,2.));
#257= NODE('44',(#255),#36,#67);
#259= CARTESIAN_POINT('45',(2.,-1.394320000000000E-9,2.));
#261= NODE('45',(#259),#36,#67);
#263= CARTESIAN_POINT('46',(3.,-4.,-7.433150000000000E-9));
#265= NODE('46',(#263),#36,#67);
#267= CARTESIAN_POINT('47',(3.,-3.,5.927840000000000E-9));
#269= NODE('47',(#267),#36,#67);
#271= CARTESIAN_POINT('48',(3.,-2.,-4.363830000000000E-8));
```

```
#273= NODE('48',(#271),#36,#67);
#275= CARTESIAN_POINT('49',(3.,-1.,-9.320450000000001E-8));
#277= NODE('49',(#275),#36,#67);
#279= CARTESIAN_POINT('50',(3.,3.434650000000000E-9,
-7.984350000000000E-8));
#281= NODE('50',(#279),#36,#67);
#283= CARTESIAN_POINT('51',(3.,-4.,1.));
#285= NODE('51',(#283),#36,#67);
#287= CARTESIAN_POINT('52',(3.,-3.,1.));
#289= NODE('52',(#287),#36,#67);
#291= CARTESIAN_POINT('53',(3.,-2.,1.));
#293= NODE('53',(#291),#36,#67);
#295= CARTESIAN_POINT('54',(3.,-1.,1.));
#297= NODE('54',(#295),#36,#67);
#299= CARTESIAN_POINT('55',(3.,2.175020000000000E-9,1.));
#301= NODE('55',(#299),#36,#67);
#303= CARTESIAN_POINT('56',(3.,-4.,2.));
#305= NODE('56',(#303),#36,#67);
#307= CARTESIAN_POINT('57',(3.,-3.,2.));
#309= NODE('57',(#307),#36,#67);
#311= CARTESIAN_POINT('58',(3.,-2.,2.));
#313= NODE('58',(#311),#36,#67);
#315= CARTESIAN_POINT('59',(3.,-1.,2.));
#317= NODE('59',(#315),#36,#67);
#319= CARTESIAN_POINT('60',(3.,9.154000000000000E-10,2.));
#321= NODE('60',(#319),#36,#67);
#323= CARTESIAN_POINT('61',(4.,-4.,1.527530000000000E-9));
#325= NODE('61',(#323),#36,#67);
#327= CARTESIAN_POINT('62',(4.,-3.,1.934220000000000E-8));
#329= NODE('62',(#327),#36,#67);
#331= CARTESIAN_POINT('63',(4.,-2.,-4.674600000000000E-8));
#333= NODE('63',(#331),#36,#67);
#335= CARTESIAN_POINT('64',(4.,-1.,-1.128340000000000E-7));
#337= NODE('64',(#335),#36,#67);
#339= CARTESIAN_POINT('65',(4.,6.584120000000000E-9,
-9.501960000000001E-8));
#341= NODE('65',(#339),#36,#67);
#343= CARTESIAN_POINT('66',(4.,-4.,1.));
#345= NODE('66',(#343),#36,#67);
#347= CARTESIAN_POINT('67',(4.,-3.,1.));
#349= NODE('67',(#347),#36,#67);
#351= CARTESIAN_POINT('68',(4.,-2.,1.));
#353= NODE('68',(#351),#36,#67);
#355= CARTESIAN_POINT('69',(4.,-1.,1.));
#357= NODE('69',(#355),#36,#67);
#359= CARTESIAN_POINT('70',(4.,4.904620000000000E-9,1.));
#361= NODE('70',(#359),#36,#67);
#363= CARTESIAN_POINT('71',(4.,-4.,2.));
#365= NODE('71',(#363),#36,#67);
#367= CARTESIAN_POINT('72',(4.,-3.,2.));
#369= NODE('72',(#367),#36,#67);
#371= CARTESIAN_POINT('73',(4.,-2.,2.));
#373= NODE('73',(#371),#36,#67);
#375= CARTESIAN_POINT('74',(4.,-1.,2.));
#377= NODE('74',(#375),#36,#67);
#379= CARTESIAN_POINT('75',(4.,3.225120000000000E-9,2.));
#381= NODE('75',(#379),#36,#67);
#383= CARTESIAN_POINT('76',(10.,-5.275770000000000E-8,0.));
#385= NODE('76',(#383),#36,#67);
#387= CARTESIAN_POINT('77',(9.,-5.275770000000000E-8,0.));
#389= NODE('77',(#387),#36,#67);
#391= CARTESIAN_POINT('78',(8.,-5.275770000000000E-8,0.));
#393= NODE('78',(#391),#36,#67);
#395= CARTESIAN_POINT('79',(7.,-5.275770000000000E-8,0.));
#397= NODE('79',(#395),#36,#67);
#399= CARTESIAN_POINT('80',(6.,-5.275770000000000E-8,0.));
#401= NODE('80',(#399),#36,#67);
#403= CARTESIAN_POINT('81',(5.,-5.275770000000000E-8,0.));
#405= NODE('81',(#403),#36,#67);
#407= CARTESIAN_POINT('83',(10.,-1.,0.));
#409= NODE('83',(#407),#36,#67);
```

```
#411= CARTESIAN_POINT('84',(9.,-1.,0.));
#413= NODE('84',(#411),#36,#67);
#415= CARTESIAN_POINT('85',(8.,-1.,0.));
#417= NODE('85',(#415),#36,#67);
#419= CARTESIAN_POINT('86',(7.,-1.,0.));
#421= NODE('86',(#419),#36,#67);
#423= CARTESIAN_POINT('87',(6.,-1.,0.));
#425= NODE('87',(#423),#36,#67);
#427= CARTESIAN_POINT('88',(5.,-1.,0.));
#429= NODE('88',(#427),#36,#67);
#431= CARTESIAN_POINT('90',(10.,-2.,0.));
#433= NODE('90',(#431),#36,#67);
#435= CARTESIAN_POINT('91',(9.,-2.,0.));
#437= NODE('91',(#435),#36,#67);
#439= CARTESIAN_POINT('92',(8.,-2.,0.));
#441= NODE('92',(#439),#36,#67);
#443= CARTESIAN_POINT('93',(7.,-2.,0.));
#445= NODE('93',(#443),#36,#67);
#447= CARTESIAN_POINT('94',(6.,-2.,0.));
#449= NODE('94',(#447),#36,#67);
#451= CARTESIAN_POINT('95',(5.,-2.,0.));
#453= NODE('95',(#451),#36,#67);
#455= CARTESIAN_POINT('97',(10.,-3.,0.));
#457= NODE('97',(#455),#36,#67);
#459= CARTESIAN_POINT('98',(9.,-3.,0.));
#461= NODE('98',(#459),#36,#67);
#463= CARTESIAN_POINT('99',(8.,-3.,0.));
#465= NODE('99',(#463),#36,#67);
#467= CARTESIAN_POINT('100',(7.,-3.,0.));
#469= NODE('100',(#467),#36,#67);
#471= CARTESIAN_POINT('101',(6.,-3.,0.));
#473= NODE('101',(#471),#36,#67);
#475= CARTESIAN_POINT('102',(5.,-3.,0.));
#477= NODE('102',(#475),#36,#67);
#479= CARTESIAN_POINT('104',(10.,-4.,0.));
#481= NODE('104',(#479),#36,#67);
#483= CARTESIAN_POINT('105',(9.,-4.,0.));
#485= NODE('105',(#483),#36,#67);
#487= CARTESIAN_POINT('106',(8.,-4.,0.));
#489= NODE('106',(#487),#36,#67);
#491= CARTESIAN_POINT('107',(7.,-4.,0.));
#493= NODE('107',(#491),#36,#67);
#495= CARTESIAN_POINT('108',(6.,-4.,0.));
#497= NODE('108',(#495),#36,#67);
#499= CARTESIAN_POINT('109',(5.,-4.,0.));
#501= NODE('109',(#499),#36,#67);
#503= CARTESIAN_POINT('111',(10.,-5.275770000000000E-8,1.));
#505= NODE('111',(#503),#36,#67);
#507= CARTESIAN_POINT('112',(9.,-5.275770000000000E-8,1.));
#509= NODE('112',(#507),#36,#67);
#511= CARTESIAN_POINT('113',(8.,-5.275770000000000E-8,1.));
#513= NODE('113',(#511),#36,#67);
#515= CARTESIAN_POINT('114',(7.,-5.275770000000000E-8,1.));
#517= NODE('114',(#515),#36,#67);
#519= CARTESIAN_POINT('115',(6.,-5.275770000000000E-8,1.));
#521= NODE('115',(#519),#36,#67);
#523= CARTESIAN_POINT('116',(5.,-5.275770000000000E-8,1.));
#525= NODE('116',(#523),#36,#67);
#527= CARTESIAN_POINT('118',(10.,-1.,1.));
#529= NODE('118',(#527),#36,#67);
#531= CARTESIAN_POINT('119',(9.,-1.,1.));
#533= NODE('119',(#531),#36,#67);
#535= CARTESIAN_POINT('120',(8.,-1.,1.));
#537= NODE('120',(#535),#36,#67);
#539= CARTESIAN_POINT('121',(7.,-1.,1.));
#541= NODE('121',(#539),#36,#67);
#543= CARTESIAN_POINT('122',(6.,-1.,1.));
#545= NODE('122',(#543),#36,#67);
#547= CARTESIAN_POINT('123',(5.,-1.,1.));
#549= NODE('123',(#547),#36,#67);
#551= CARTESIAN_POINT('125',(10.,-2.,1.));
```

```
#553= NODE('125',(#551),#36,#67);
#555= CARTESIAN_POINT('126',(9.,-2.,1.));
#557= NODE('126',(#555),#36,#67);
#559= CARTESIAN_POINT('127',(8.,-2.,1.));
#561= NODE('127',(#559),#36,#67);
#563= CARTESIAN_POINT('128',(7.,-2.,1.));
#565= NODE('128',(#563),#36,#67);
#567= CARTESIAN_POINT('129',(6.,-2.,1.));
#569= NODE('129',(#567),#36,#67);
#571= CARTESIAN_POINT('130',(5.,-2.,1.));
#573= NODE('130',(#571),#36,#67);
#575= CARTESIAN_POINT('132',(10.,-3.,1.));
#577= NODE('132',(#575),#36,#67);
#579= CARTESIAN_POINT('133',(9.,-3.,1.));
#581= NODE('133',(#579),#36,#67);
#583= CARTESIAN_POINT('134',(8.,-3.,1.));
#585= NODE('134',(#583),#36,#67);
#587= CARTESIAN_POINT('135',(7.,-3.,1.));
#589= NODE('135',(#587),#36,#67);
#591= CARTESIAN_POINT('136',(6.,-3.,1.));
#593= NODE('136',(#591),#36,#67);
#595= CARTESIAN_POINT('137',(5.,-3.,1.));
#597= NODE('137',(#595),#36,#67);
#599= CARTESIAN_POINT('139',(10.,-4.,1.));
#601= NODE('139',(#599),#36,#67);
#603= CARTESIAN_POINT('140',(9.,-4.,1.));
#605= NODE('140',(#603),#36,#67);
#607= CARTESIAN_POINT('141',(8.,-4.,1.));
#609= NODE('141',(#607),#36,#67);
#611= CARTESIAN_POINT('142',(7.,-4.,1.));
#613= NODE('142',(#611),#36,#67);
#615= CARTESIAN_POINT('143',(6.,-4.,1.));
#617= NODE('143',(#615),#36,#67);
#619= CARTESIAN_POINT('144',(5.,-4.,1.));
#621= NODE('144',(#619),#36,#67);
#623= CARTESIAN_POINT('146',(10.,-5.275770000000000E-8,2.));
#625= NODE('146',(#623),#36,#67);
#627= CARTESIAN_POINT('147',(9.,-5.275770000000000E-8,2.));
#629= NODE('147',(#627),#36,#67);
#631= CARTESIAN_POINT('148',(8.,-5.275770000000000E-8,2.));
#633= NODE('148',(#631),#36,#67);
#635= CARTESIAN_POINT('149',(7.,-5.275770000000000E-8,2.));
#637= NODE('149',(#635),#36,#67);
#639= CARTESIAN_POINT('150',(6.,-5.275770000000000E-8,2.));
#641= NODE('150',(#639),#36,#67);
#643= CARTESIAN_POINT('151',(5.,-5.275770000000000E-8,2.));
#645= NODE('151',(#643),#36,#67);
#647= CARTESIAN_POINT('153',(10.,-1.,2.));
#649= NODE('153',(#647),#36,#67);
#651= CARTESIAN_POINT('154',(9.,-1.,2.));
#653= NODE('154',(#651),#36,#67);
#655= CARTESIAN_POINT('155',(8.,-1.,2.));
#657= NODE('155',(#655),#36,#67);
#659= CARTESIAN_POINT('156',(7.,-1.,2.));
#661= NODE('156',(#659),#36,#67);
#663= CARTESIAN_POINT('157',(6.,-1.,2.));
#665= NODE('157',(#663),#36,#67);
#667= CARTESIAN_POINT('158',(5.,-1.,2.));
#669= NODE('158',(#667),#36,#67);
#671= CARTESIAN_POINT('160',(10.,-2.,2.));
#673= NODE('160',(#671),#36,#67);
#675= CARTESIAN_POINT('161',(9.,-2.,2.));
#677= NODE('161',(#675),#36,#67);
#679= CARTESIAN_POINT('162',(8.,-2.,2.));
#681= NODE('162',(#679),#36,#67);
#683= CARTESIAN_POINT('163',(7.,-2.,2.));
#685= NODE('163',(#683),#36,#67);
#687= CARTESIAN_POINT('164',(6.,-2.,2.));
#689= NODE('164',(#687),#36,#67);
#691= CARTESIAN_POINT('165',(5.,-2.,2.));
#693= NODE('165',(#691),#36,#67);
```

```
#695= CARTESIAN_POINT('167',(10.,-3.,2.));
#697= NODE('167',(#695),#36,#67);
#699= CARTESIAN_POINT('168',(9.,-3.,2.));
#701= NODE('168',(#699),#36,#67);
#703= CARTESIAN_POINT('169',(8.,-3.,2.));
#705= NODE('169',(#703),#36,#67);
#707= CARTESIAN_POINT('170',(7.,-3.,2.));
#709= NODE('170',(#707),#36,#67);
#711= CARTESIAN_POINT('171',(6.,-3.,2.));
#713= NODE('171',(#711),#36,#67);
#715= CARTESIAN_POINT('172',(5.,-3.,2.));
#717= NODE('172',(#715),#36,#67);
#719= CARTESIAN_POINT('174',(10.,-4.,2.));
#721= NODE('174',(#719),#36,#67);
#723= CARTESIAN_POINT('175',(9.,-4.,2.));
#725= NODE('175',(#723),#36,#67);
#727= CARTESIAN_POINT('176',(8.,-4.,2.));
#729= NODE('176',(#727),#36,#67);
#731= CARTESIAN_POINT('177',(7.,-4.,2.));
#733= NODE('177',(#731),#36,#67);
#735= CARTESIAN_POINT('178',(6.,-4.,2.));
#737= NODE('178',(#735),#36,#67);
#739= CARTESIAN_POINT('179',(5.,-4.,2.));
#741= NODE('179',(#739),#36,#67);
#743= CARTESIAN_POINT('181',(16.,-5.275770000000000E-8,0.));
#745= NODE('181',(#743),#36,#67);
#747= CARTESIAN_POINT('182',(15.,-5.275770000000000E-8,0.));
#749= NODE('182',(#747),#36,#67);
#751= CARTESIAN_POINT('183',(14.,-5.275770000000000E-8,0.));
#753= NODE('183',(#751),#36,#67);
#755= CARTESIAN_POINT('184',(13.,-5.275770000000000E-8,0.));
#757= NODE('184',(#755),#36,#67);
#759= CARTESIAN_POINT('185',(12.,-5.275770000000000E-8,0.));
#761= NODE('185',(#759),#36,#67);
#763= CARTESIAN_POINT('186',(11.,-5.275770000000000E-8,0.));
#765= NODE('186',(#763),#36,#67);
#767= CARTESIAN_POINT('188',(16.,-1.,0.));
#769= NODE('188',(#767),#36,#67);
#771= CARTESIAN_POINT('189',(15.,-1.,0.));
#773= NODE('189',(#771),#36,#67);
#775= CARTESIAN_POINT('190',(14.,-1.,0.));
#777= NODE('190',(#775),#36,#67);
#779= CARTESIAN_POINT('191',(13.,-1.,0.));
#781= NODE('191',(#779),#36,#67);
#783= CARTESIAN_POINT('192',(12.,-1.,0.));
#785= NODE('192',(#783),#36,#67);
#787= CARTESIAN_POINT('193',(11.,-1.,0.));
#789= NODE('193',(#787),#36,#67);
#791= CARTESIAN_POINT('195',(16.,-2.,0.));
#793= NODE('195',(#791),#36,#67);
#795= CARTESIAN_POINT('196',(15.,-2.,0.));
#797= NODE('196',(#795),#36,#67);
#799= CARTESIAN_POINT('197',(14.,-2.,0.));
#801= NODE('197',(#799),#36,#67);
#803= CARTESIAN_POINT('198',(13.,-2.,0.));
#805= NODE('198',(#803),#36,#67);
#807= CARTESIAN_POINT('199',(12.,-2.,0.));
#809= NODE('199',(#807),#36,#67);
#811= CARTESIAN_POINT('200',(11.,-2.,0.));
#813= NODE('200',(#811),#36,#67);
#815= CARTESIAN_POINT('202',(16.,-3.,0.));
#817= NODE('202',(#815),#36,#67);
#819= CARTESIAN_POINT('203',(15.,-3.,0.));
#821= NODE('203',(#819),#36,#67);
#823= CARTESIAN_POINT('204',(14.,-3.,0.));
#825= NODE('204',(#823),#36,#67);
#827= CARTESIAN_POINT('205',(13.,-3.,0.));
#829= NODE('205',(#827),#36,#67);
#831= CARTESIAN_POINT('206',(12.,-3.,0.));
#833= NODE('206',(#831),#36,#67);
#835= CARTESIAN_POINT('207',(11.,-3.,0.));
```

```
#837= NODE('207',(#835),#36,#67);
#839= CARTESIAN_POINT('209',(16.,-4.,0.));
#841= NODE('209',(#839),#36,#67);
#843= CARTESIAN_POINT('210',(15.,-4.,0.));
#845= NODE('210',(#843),#36,#67);
#847= CARTESIAN_POINT('211',(14.,-4.,0.));
#849= NODE('211',(#847),#36,#67);
#851= CARTESIAN_POINT('212',(13.,-4.,0.));
#853= NODE('212',(#851),#36,#67);
#855= CARTESIAN_POINT('213',(12.,-4.,0.));
#857= NODE('213',(#855),#36,#67);
#859= CARTESIAN_POINT('214',(11.,-4.,0.));
#861= NODE('214',(#859),#36,#67);
#863= CARTESIAN_POINT('216',(16.,-5.275770000000000E-8,1.));
#865= NODE('216',(#863),#36,#67);
#867= CARTESIAN_POINT('217',(15.,-5.275770000000000E-8,1.));
#869= NODE('217',(#867),#36,#67);
#871= CARTESIAN_POINT('218',(14.,-5.275770000000000E-8,1.));
#873= NODE('218',(#871),#36,#67);
#875= CARTESIAN_POINT('219',(13.,-5.275770000000000E-8,1.));
#877= NODE('219',(#875),#36,#67);
#879= CARTESIAN_POINT('220',(12.,-5.275770000000000E-8,1.));
#881= NODE('220',(#879),#36,#67);
#883= CARTESIAN_POINT('221',(11.,-5.275770000000000E-8,1.));
#885= NODE('221',(#883),#36,#67);
#887= CARTESIAN_POINT('223',(16.,-1.,1.));
#889= NODE('223',(#887),#36,#67);
#891= CARTESIAN_POINT('224',(15.,-1.,1.));
#893= NODE('224',(#891),#36,#67);
#895= CARTESIAN_POINT('225',(14.,-1.,1.));
#897= NODE('225',(#895),#36,#67);
#899= CARTESIAN_POINT('226',(13.,-1.,1.));
#901= NODE('226',(#899),#36,#67);
#903= CARTESIAN_POINT('227',(12.,-1.,1.));
#905= NODE('227',(#903),#36,#67);
#907= CARTESIAN_POINT('228',(11.,-1.,1.));
#909= NODE('228',(#907),#36,#67);
#911= CARTESIAN_POINT('230',(16.,-2.,1.));
#913= NODE('230',(#911),#36,#67);
#915= CARTESIAN_POINT('231',(15.,-2.,1.));
#917= NODE('231',(#915),#36,#67);
#919= CARTESIAN_POINT('232',(14.,-2.,1.));
#921= NODE('232',(#919),#36,#67);
#923= CARTESIAN_POINT('233',(13.,-2.,1.));
#925= NODE('233',(#923),#36,#67);
#927= CARTESIAN_POINT('234',(12.,-2.,1.));
#929= NODE('234',(#927),#36,#67);
#931= CARTESIAN_POINT('235',(11.,-2.,1.));
#933= NODE('235',(#931),#36,#67);
#935= CARTESIAN_POINT('237',(16.,-3.,1.));
#937= NODE('237',(#935),#36,#67);
#939= CARTESIAN_POINT('238',(15.,-3.,1.));
#941= NODE('238',(#939),#36,#67);
#943= CARTESIAN_POINT('239',(14.,-3.,1.));
#945= NODE('239',(#943),#36,#67);
#947= CARTESIAN_POINT('240',(13.,-3.,1.));
#949= NODE('240',(#947),#36,#67);
#951= CARTESIAN_POINT('241',(12.,-3.,1.));
#953= NODE('241',(#951),#36,#67);
#955= CARTESIAN_POINT('242',(11.,-3.,1.));
#957= NODE('242',(#955),#36,#67);
#959= CARTESIAN_POINT('244',(16.,-4.,1.));
#961= NODE('244',(#959),#36,#67);
#963= CARTESIAN_POINT('245',(15.,-4.,1.));
#965= NODE('245',(#963),#36,#67);
#967= CARTESIAN_POINT('246',(14.,-4.,1.));
#969= NODE('246',(#967),#36,#67);
#971= CARTESIAN_POINT('247',(13.,-4.,1.));
#973= NODE('247',(#971),#36,#67);
#975= CARTESIAN_POINT('248',(12.,-4.,1.));
#977= NODE('248',(#975),#36,#67);
```

```
#979= CARTESIAN_POINT('249',(11.,-4.,1.));
#981= NODE('249',(#979),#36,#67);
#983= CARTESIAN_POINT('251',(16.,-5.275770000000000E-8,2.));
#985= NODE('251',(#983),#36,#67);
#987= CARTESIAN_POINT('252',(15.,-5.275770000000000E-8,2.));
#989= NODE('252',(#987),#36,#67);
#991= CARTESIAN_POINT('253',(14.,-5.275770000000000E-8,2.));
#993= NODE('253',(#991),#36,#67);
#995= CARTESIAN_POINT('254',(13.,-5.275770000000000E-8,2.));
#997= NODE('254',(#995),#36,#67);
#999= CARTESIAN_POINT('255',(12.,-5.275770000000000E-8,2.));
#1001= NODE('255',(#999),#36,#67);
#1003= CARTESIAN_POINT('256',(11.,-5.275770000000000E-8,2.));
#1005= NODE('256',(#1003),#36,#67);
#1007= CARTESIAN_POINT('258',(16.,-1.,2.));
#1009= NODE('258',(#1007),#36,#67);
#1011= CARTESIAN_POINT('259',(15.,-1.,2.));
#1013= NODE('259',(#1011),#36,#67);
#1015= CARTESIAN_POINT('260',(14.,-1.,2.));
#1017= NODE('260',(#1015),#36,#67);
#1019= CARTESIAN_POINT('261',(13.,-1.,2.));
#1021= NODE('261',(#1019),#36,#67);
#1023= CARTESIAN_POINT('262',(12.,-1.,2.));
#1025= NODE('262',(#1023),#36,#67);
#1027= CARTESIAN_POINT('263',(11.,-1.,2.));
#1029= NODE('263',(#1027),#36,#67);
#1031= CARTESIAN_POINT('265',(16.,-2.,2.));
#1033= NODE('265',(#1031),#36,#67);
#1035= CARTESIAN_POINT('266',(15.,-2.,2.));
#1037= NODE('266',(#1035),#36,#67);
#1039= CARTESIAN_POINT('267',(14.,-2.,2.));
#1041= NODE('267',(#1039),#36,#67);
#1043= CARTESIAN_POINT('268',(13.,-2.,2.));
#1045= NODE('268',(#1043),#36,#67);
#1047= CARTESIAN_POINT('269',(12.,-2.,2.));
#1049= NODE('269',(#1047),#36,#67);
#1051= CARTESIAN_POINT('270',(11.,-2.,2.));
#1053= NODE('270',(#1051),#36,#67);
#1055= CARTESIAN_POINT('272',(16.,-3.,2.));
#1057= NODE('272',(#1055),#36,#67);
#1059= CARTESIAN_POINT('273',(15.,-3.,2.));
#1061= NODE('273',(#1059),#36,#67);
#1063= CARTESIAN_POINT('274',(14.,-3.,2.));
#1065= NODE('274',(#1063),#36,#67);
#1067= CARTESIAN_POINT('275',(13.,-3.,2.));
#1069= NODE('275',(#1067),#36,#67);
#1071= CARTESIAN_POINT('276',(12.,-3.,2.));
#1073= NODE('276',(#1071),#36,#67);
#1075= CARTESIAN_POINT('277',(11.,-3.,2.));
#1077= NODE('277',(#1075),#36,#67);
#1079= CARTESIAN_POINT('279',(16.,-4.,2.));
#1081= NODE('279',(#1079),#36,#67);
#1083= CARTESIAN_POINT('280',(15.,-4.,2.));
#1085= NODE('280',(#1083),#36,#67);
#1087= CARTESIAN_POINT('281',(14.,-4.,2.));
#1089= NODE('281',(#1087),#36,#67);
#1091= CARTESIAN_POINT('282',(13.,-4.,2.));
#1093= NODE('282',(#1091),#36,#67);
#1095= CARTESIAN_POINT('283',(12.,-4.,2.));
#1097= NODE('283',(#1095),#36,#67);
#1099= CARTESIAN_POINT('284',(11.,-4.,2.));
#1101= NODE('284',(#1099),#36,#67);
#1103= ELEMENT_MATERIAL('MAT1.1','Fea Material',(#1115,#1123,#1129));
#1104= MEASURE_REPRESENTATION_ITEM('representation_item_name_1',
THERMODYNAMIC_TEMPERATURE_MEASURE(70.),#10);
#1105= PROPERTY_DEFINITION_REPRESENTATION(#1106,#1108);
#1106= PROPERTY_DEFINITION('Material Property Definition',$,#1107);
#1107= CHARACTERIZED_OBJECT('property',$);
#1108= REPRESENTATION('representation_id_1',(#1104),#36);
#1109= ID_ATTRIBUTE('MAT1.1',#1106);
#1111= DATA_ENVIRONMENT('DATA_ENV','Property_conditions',(#1105));
```

```
#1113= FEA_LINEAR_ELASTICITY('',FEA_ISOTROPIC_SYMMETRIC_TENSOR4_3D((
10000000.,0.33)));
#1115= FEA_MATERIAL_PROPERTY_REPRESENTATION(#1116,#1118,#1111);
#1116= MATERIAL_PROPERTY('MAT1.1.0',$,#1117);
#1117= CHARACTERIZED_OBJECT('material_property',$);
#1118= REPRESENTATION('representation_id_2',(#1113),#36);
#1119= DESCRIPTION_ATTRIBUTE('Elasticity',#1115);
#1122= FEA_MASS_DENSITY('',0.000254);
#1123= FEA_MATERIAL_PROPERTY_REPRESENTATION(#1124,#1125,#1111);
#1124= MATERIAL_PROPERTY('MAT1.1.1',$,#1117);
#1125= REPRESENTATION('representation_id_3',(#1122),#36);
#1126= DESCRIPTION_ATTRIBUTE('MassDensity',#1123);
#1128= FEA_TANGENTIAL_COEFFICIENT_OF_LINEAR_THERMAL_EXPANSION('',
ISOTROPIC_SYMMETRIC_TENSOR2_3D(0.));
#1129= FEA_MATERIAL_PROPERTY_REPRESENTATION(#1130,#1131,#1111);
#1130= MATERIAL_PROPERTY('MAT1.1.2',$,#1117);
#1131= REPRESENTATION('representation_id_4',(#1128),#36);
#1132= DESCRIPTION_ATTRIBUTE('TangentCTE',#1129);
#1134= VOLUME_3D_ELEMENT_REPRESENTATION('1',(#1143),#1135,(#80,#89,#109,
#105,#145,#149,#169,#165),#67,#1145,#1103);
#1135=
/* GEOMETRIC_REPRESENTATION_CONTEXT+PARAMETRIC_REPRESENTATION_CONTEXT */
(GEOMETRIC_REPRESENTATION_CONTEXT(3)PARAMETRIC_REPRESENTATION_CONTEXT()
REPRESENTATION_CONTEXT('REP_CONTEXT_FEA','3D'));
#1137= REPRESENTATION('Default SimDM Fea representation',(#1139),#1135);
#1139= REPRESENTATION_ITEM('Default SimDM item');
#1140= ID_ATTRIBUTE('Default SimDM Fea representation',#1137);
#1141= DESCRIPTION_ATTRIBUTE('Default SimDM Fea representation',#1137);
#1143= ARBITRARY_VOLUME_3D_ELEMENT_COORDINATE_SYSTEM('',#35);
#1145= VOLUME_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,
'LINEAR_HEXAHEDRON.CHEXA',(ENUMERATED_VOLUME_ELEMENT_PURPOSE(
.STRESS_DISPLACEMENT.)),.HEXAHEDRON.);
#1147= VOLUME_3D_ELEMENT_REPRESENTATION('2',(#1143),#1135,(#89,#93,#113,
#109,#149,#153,#173,#169),#67,#1145,#1103);
#1150= VOLUME_3D_ELEMENT_REPRESENTATION('3',(#1143),#1135,(#93,#97,#117,
#113,#153,#157,#177,#173),#67,#1145,#1103);
#1153= VOLUME_3D_ELEMENT_REPRESENTATION('4',(#1143),#1135,(#97,#101,#121
,#117,#157,#161,#181,#177),#67,#1145,#1103);
#1156= VOLUME_3D_ELEMENT_REPRESENTATION('5',(#1143),#1135,(#105,#109,
#129,#125,#165,#169,#189,#185),#67,#1145,#1103);
#1159= VOLUME_3D_ELEMENT_REPRESENTATION('6',(#1143),#1135,(#109,#113,
#133,#129,#169,#173,#193,#189),#67,#1145,#1103);
#1162= VOLUME_3D_ELEMENT_REPRESENTATION('7',(#1143),#1135,(#113,#117,
#137,#133,#173,#177,#197,#193),#67,#1145,#1103);
#1165= VOLUME_3D_ELEMENT_REPRESENTATION('8',(#1143),#1135,(#117,#121,
#141,#137,#177,#181,#201,#197),#67,#1145,#1103);
#1168= VOLUME_3D_ELEMENT_REPRESENTATION('9',(#1143),#1135,(#145,#149,
#169,#165,#205,#209,#229,#225),#67,#1145,#1103);
#1171= VOLUME_3D_ELEMENT_REPRESENTATION('10',(#1143),#1135,(#149,#153,
#173,#169,#209,#213,#233,#229),#67,#1145,#1103);
#1174= VOLUME_3D_ELEMENT_REPRESENTATION('11',(#1143),#1135,(#153,#157,
#177,#173,#213,#217,#237,#233),#67,#1145,#1103);
#1177= VOLUME_3D_ELEMENT_REPRESENTATION('12',(#1143),#1135,(#157,#161,
#181,#177,#217,#221,#241,#237),#67,#1145,#1103);
#1180= VOLUME_3D_ELEMENT_REPRESENTATION('13',(#1143),#1135,(#165,#169,
#189,#185,#225,#229,#249,#245),#67,#1145,#1103);
#1183= VOLUME_3D_ELEMENT_REPRESENTATION('14',(#1143),#1135,(#169,#173,
#193,#189,#229,#233,#253,#249),#67,#1145,#1103);
#1186= VOLUME_3D_ELEMENT_REPRESENTATION('15',(#1143),#1135,(#173,#177,
#197,#193,#233,#237,#257,#253),#67,#1145,#1103);
#1189= VOLUME_3D_ELEMENT_REPRESENTATION('16',(#1143),#1135,(#177,#181,
#201,#197,#237,#241,#261,#257),#67,#1145,#1103);
#1192= VOLUME_3D_ELEMENT_REPRESENTATION('17',(#1143),#1135,(#205,#209,
#229,#225,#265,#269,#289,#285),#67,#1145,#1103);
#1195= VOLUME_3D_ELEMENT_REPRESENTATION('18',(#1143),#1135,(#209,#213,
#233,#229,#269,#273,#293,#289),#67,#1145,#1103);
#1198= VOLUME_3D_ELEMENT_REPRESENTATION('19',(#1143),#1135,(#213,#217,
#237,#233,#273,#277,#297,#293),#67,#1145,#1103);
#1201= VOLUME_3D_ELEMENT_REPRESENTATION('20',(#1143),#1135,(#217,#221,
#241,#237,#277,#281,#301,#297),#67,#1145,#1103);
#1204= VOLUME_3D_ELEMENT_REPRESENTATION('21',(#1143),#1135,(#225,#229,
```

```
#249,#245,#285,#289,#309,#305),#67,#1145,#1103);
#1207= VOLUME_3D_ELEMENT_REPRESENTATION('22',(#1143),#1135,(#229,#233,
#253,#249,#289,#293,#313,#309),#67,#1145,#1103);
#1210= VOLUME_3D_ELEMENT_REPRESENTATION('23',(#1143),#1135,(#233,#237,
#257,#253,#293,#297,#317,#313),#67,#1145,#1103);
#1213= VOLUME_3D_ELEMENT_REPRESENTATION('24',(#1143),#1135,(#237,#241,
#261,#257,#297,#301,#321,#317),#67,#1145,#1103);
#1216= VOLUME_3D_ELEMENT_REPRESENTATION('25',(#1143),#1135,(#265,#269,
#289,#285,#325,#329,#349,#345),#67,#1145,#1103);
#1219= VOLUME_3D_ELEMENT_REPRESENTATION('26',(#1143),#1135,(#269,#273,
#293,#289,#329,#333,#353,#349),#67,#1145,#1103);
#1222= VOLUME_3D_ELEMENT_REPRESENTATION('27',(#1143),#1135,(#273,#277,
#297,#293,#333,#337,#357,#353),#67,#1145,#1103);
#1225= VOLUME_3D_ELEMENT_REPRESENTATION('28',(#1143),#1135,(#277,#281,
#301,#297,#337,#341,#361,#357),#67,#1145,#1103);
#1228= VOLUME_3D_ELEMENT_REPRESENTATION('29',(#1143),#1135,(#285,#289,
#309,#305,#345,#349,#369,#365),#67,#1145,#1103);
#1231= VOLUME_3D_ELEMENT_REPRESENTATION('30',(#1143),#1135,(#289,#293,
#313,#309,#349,#353,#373,#369),#67,#1145,#1103);
#1234= VOLUME_3D_ELEMENT_REPRESENTATION('31',(#1143),#1135,(#293,#297,
#317,#313,#353,#357,#377,#373),#67,#1145,#1103);
#1237= VOLUME_3D_ELEMENT_REPRESENTATION('32',(#1143),#1135,(#297,#301,
#321,#317,#357,#361,#381,#377),#67,#1145,#1103);
#1240= VOLUME_3D_ELEMENT_REPRESENTATION('33',(#1143),#1135,(#385,#409,
#505,#389),#67,#1243,#1103);
#1243= VOLUME_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,
'LINEAR_TETRAHEDRON.CTETRA',(ENUMERATED_VOLUME_ELEMENT_PURPOSE(
.STRESS_DISPLACEMENT.)),.TETRAHEDRON.);
#1245= VOLUME_3D_ELEMENT_REPRESENTATION('34',(#1143),#1135,(#529,#505,
#409,#533),#67,#1243,#1103);
#1248= VOLUME_3D_ELEMENT_REPRESENTATION('35',(#1143),#1135,(#413,#389,
#533,#409),#67,#1243,#1103);
#1251= VOLUME_3D_ELEMENT_REPRESENTATION('36',(#1143),#1135,(#509,#533,
#389,#505),#67,#1243,#1103);
#1254= VOLUME_3D_ELEMENT_REPRESENTATION('37',(#1143),#1135,(#389,#409,
#505,#533),#67,#1243,#1103);
#1257= VOLUME_3D_ELEMENT_REPRESENTATION('38',(#1143),#1135,(#413,#533,
#389,#417),#67,#1243,#1103);
#1260= VOLUME_3D_ELEMENT_REPRESENTATION('39',(#1143),#1135,(#509,#389,
#533,#513),#67,#1243,#1103);
#1263= VOLUME_3D_ELEMENT_REPRESENTATION('40',(#1143),#1135,(#393,#513,
#417,#389),#67,#1243,#1103);
#1266= VOLUME_3D_ELEMENT_REPRESENTATION('41',(#1143),#1135,(#537,#417,
#513,#533),#67,#1243,#1103);
#1269= VOLUME_3D_ELEMENT_REPRESENTATION('42',(#1143),#1135,(#389,#417,
#533,#513),#67,#1243,#1103);
#1272= VOLUME_3D_ELEMENT_REPRESENTATION('43',(#1143),#1135,(#393,#417,
#513,#397),#67,#1243,#1103);
#1275= VOLUME_3D_ELEMENT_REPRESENTATION('44',(#1143),#1135,(#537,#513,
#417,#541),#67,#1243,#1103);
#1278= VOLUME_3D_ELEMENT_REPRESENTATION('45',(#1143),#1135,(#421,#397,
#541,#417),#67,#1243,#1103);
#1281= VOLUME_3D_ELEMENT_REPRESENTATION('46',(#1143),#1135,(#517,#541,
#397,#513),#67,#1243,#1103);
#1284= VOLUME_3D_ELEMENT_REPRESENTATION('47',(#1143),#1135,(#397,#417,
#513,#541),#67,#1243,#1103);
#1287= VOLUME_3D_ELEMENT_REPRESENTATION('48',(#1143),#1135,(#421,#541,
#397,#425),#67,#1243,#1103);
#1290= VOLUME_3D_ELEMENT_REPRESENTATION('49',(#1143),#1135,(#517,#397,
#541,#521),#67,#1243,#1103);
#1293= VOLUME_3D_ELEMENT_REPRESENTATION('50',(#1143),#1135,(#401,#521,
#425,#397),#67,#1243,#1103);
#1296= VOLUME_3D_ELEMENT_REPRESENTATION('51',(#1143),#1135,(#545,#425,
#521,#541),#67,#1243,#1103);
#1299= VOLUME_3D_ELEMENT_REPRESENTATION('52',(#1143),#1135,(#397,#425,
#541,#521),#67,#1243,#1103);
#1302= VOLUME_3D_ELEMENT_REPRESENTATION('53',(#1143),#1135,(#401,#425,
#521,#405),#67,#1243,#1103);
#1305= VOLUME_3D_ELEMENT_REPRESENTATION('54',(#1143),#1135,(#545,#521,
#425,#549),#67,#1243,#1103);
#1308= VOLUME_3D_ELEMENT_REPRESENTATION('55',(#1143),#1135,(#429,#405,
```

```
#549,#425),#67,#1243,#1103);
#1311= VOLUME_3D_ELEMENT_REPRESENTATION('56',(#1143),#1135,(#525,#549,
#405,#521),#67,#1243,#1103);
#1314= VOLUME_3D_ELEMENT_REPRESENTATION('57',(#1143),#1135,(#405,#425,
#521,#549),#67,#1243,#1103);
#1317= VOLUME_3D_ELEMENT_REPRESENTATION('58',(#1143),#1135,(#429,#549,
#405,#337),#67,#1243,#1103);
#1320= VOLUME_3D_ELEMENT_REPRESENTATION('59',(#1143),#1135,(#525,#405,
#549,#361),#67,#1243,#1103);
#1323= VOLUME_3D_ELEMENT_REPRESENTATION('60',(#1143),#1135,(#341,#361,
#337,#405),#67,#1243,#1103);
#1326= VOLUME_3D_ELEMENT_REPRESENTATION('61',(#1143),#1135,(#357,#337,
#361,#549),#67,#1243,#1103);
#1329= VOLUME_3D_ELEMENT_REPRESENTATION('62',(#1143),#1135,(#405,#337,
#549,#361),#67,#1243,#1103);
#1332= VOLUME_3D_ELEMENT_REPRESENTATION('63',(#1143),#1135,(#433,#553,
#409,#437),#67,#1243,#1103);
#1335= VOLUME_3D_ELEMENT_REPRESENTATION('64',(#1143),#1135,(#529,#409,
#553,#533),#67,#1243,#1103);
#1338= VOLUME_3D_ELEMENT_REPRESENTATION('65',(#1143),#1135,(#413,#533,
#437,#409),#67,#1243,#1103);
#1341= VOLUME_3D_ELEMENT_REPRESENTATION('66',(#1143),#1135,(#557,#437,
#533,#553),#67,#1243,#1103);
#1344= VOLUME_3D_ELEMENT_REPRESENTATION('67',(#1143),#1135,(#409,#437,
#553,#533),#67,#1243,#1103);
#1347= VOLUME_3D_ELEMENT_REPRESENTATION('68',(#1143),#1135,(#413,#437,
#533,#417),#67,#1243,#1103);
#1350= VOLUME_3D_ELEMENT_REPRESENTATION('69',(#1143),#1135,(#557,#533,
#437,#561),#67,#1243,#1103);
#1353= VOLUME_3D_ELEMENT_REPRESENTATION('70',(#1143),#1135,(#441,#417,
#561,#437),#67,#1243,#1103);
#1356= VOLUME_3D_ELEMENT_REPRESENTATION('71',(#1143),#1135,(#537,#561,
#417,#533),#67,#1243,#1103);
#1359= VOLUME_3D_ELEMENT_REPRESENTATION('72',(#1143),#1135,(#417,#437,
#533,#561),#67,#1243,#1103);
#1362= VOLUME_3D_ELEMENT_REPRESENTATION('73',(#1143),#1135,(#441,#561,
#417,#445),#67,#1243,#1103);
#1365= VOLUME_3D_ELEMENT_REPRESENTATION('74',(#1143),#1135,(#537,#417,
#561,#541),#67,#1243,#1103);
#1368= VOLUME_3D_ELEMENT_REPRESENTATION('75',(#1143),#1135,(#421,#541,
#445,#417),#67,#1243,#1103);
#1371= VOLUME_3D_ELEMENT_REPRESENTATION('76',(#1143),#1135,(#565,#445,
#541,#561),#67,#1243,#1103);
#1374= VOLUME_3D_ELEMENT_REPRESENTATION('77',(#1143),#1135,(#417,#445,
#561,#541),#67,#1243,#1103);
#1377= VOLUME_3D_ELEMENT_REPRESENTATION('78',(#1143),#1135,(#421,#445,
#541,#425),#67,#1243,#1103);
#1380= VOLUME_3D_ELEMENT_REPRESENTATION('79',(#1143),#1135,(#565,#541,
#445,#569),#67,#1243,#1103);
#1383= VOLUME_3D_ELEMENT_REPRESENTATION('80',(#1143),#1135,(#449,#425,
#569,#445),#67,#1243,#1103);
#1386= VOLUME_3D_ELEMENT_REPRESENTATION('81',(#1143),#1135,(#545,#569,
#425,#541),#67,#1243,#1103);
#1389= VOLUME_3D_ELEMENT_REPRESENTATION('82',(#1143),#1135,(#425,#445,
#541,#569),#67,#1243,#1103);
#1392= VOLUME_3D_ELEMENT_REPRESENTATION('83',(#1143),#1135,(#449,#569,
#425,#453),#67,#1243,#1103);
#1395= VOLUME_3D_ELEMENT_REPRESENTATION('84',(#1143),#1135,(#545,#425,
#569,#549),#67,#1243,#1103);
#1398= VOLUME_3D_ELEMENT_REPRESENTATION('85',(#1143),#1135,(#429,#549,
#453,#425),#67,#1243,#1103);
#1401= VOLUME_3D_ELEMENT_REPRESENTATION('86',(#1143),#1135,(#573,#453,
#549,#569),#67,#1243,#1103);
#1404= VOLUME_3D_ELEMENT_REPRESENTATION('87',(#1143),#1135,(#425,#453,
#569,#549),#67,#1243,#1103);
#1407= VOLUME_3D_ELEMENT_REPRESENTATION('88',(#1143),#1135,(#429,#453,
#549,#337),#67,#1243,#1103);
#1410= VOLUME_3D_ELEMENT_REPRESENTATION('89',(#1143),#1135,(#573,#549,
#453,#353),#67,#1243,#1103);
#1413= VOLUME_3D_ELEMENT_REPRESENTATION('90',(#1143),#1135,(#333,#337,
#353,#453),#67,#1243,#1103);
```

```
#1416= VOLUME_3D_ELEMENT_REPRESENTATION('91',(#1143),#1135,(#357,#353,
#337,#549),#67,#1243,#1103);
#1419= VOLUME_3D_ELEMENT_REPRESENTATION('92',(#1143),#1135,(#337,#453,
#549,#353),#67,#1243,#1103);
#1422= VOLUME_3D_ELEMENT_REPRESENTATION('93',(#1143),#1135,(#433,#457,
#553,#437),#67,#1243,#1103);
#1425= VOLUME_3D_ELEMENT_REPRESENTATION('94',(#1143),#1135,(#577,#553,
#457,#581),#67,#1243,#1103);
#1428= VOLUME_3D_ELEMENT_REPRESENTATION('95',(#1143),#1135,(#461,#437,
#581,#457),#67,#1243,#1103);
#1431= VOLUME_3D_ELEMENT_REPRESENTATION('96',(#1143),#1135,(#557,#581,
#437,#553),#67,#1243,#1103);
#1434= VOLUME_3D_ELEMENT_REPRESENTATION('97',(#1143),#1135,(#437,#457,
#553,#581),#67,#1243,#1103);
#1437= VOLUME_3D_ELEMENT_REPRESENTATION('98',(#1143),#1135,(#461,#581,
#437,#465),#67,#1243,#1103);
#1440= VOLUME_3D_ELEMENT_REPRESENTATION('99',(#1143),#1135,(#557,#437,
#581,#561),#67,#1243,#1103);
#1443= VOLUME_3D_ELEMENT_REPRESENTATION('100',(#1143),#1135,(#441,#561,
#465,#437),#67,#1243,#1103);
#1446= VOLUME_3D_ELEMENT_REPRESENTATION('101',(#1143),#1135,(#585,#465,
#561,#581),#67,#1243,#1103);
#1449= VOLUME_3D_ELEMENT_REPRESENTATION('102',(#1143),#1135,(#437,#465,
#581,#561),#67,#1243,#1103);
#1452= VOLUME_3D_ELEMENT_REPRESENTATION('103',(#1143),#1135,(#441,#465,
#561,#445),#67,#1243,#1103);
#1455= VOLUME_3D_ELEMENT_REPRESENTATION('104',(#1143),#1135,(#585,#561,
#465,#589),#67,#1243,#1103);
#1458= VOLUME_3D_ELEMENT_REPRESENTATION('105',(#1143),#1135,(#469,#445,
#589,#465),#67,#1243,#1103);
#1461= VOLUME_3D_ELEMENT_REPRESENTATION('106',(#1143),#1135,(#565,#589,
#445,#561),#67,#1243,#1103);
#1464= VOLUME_3D_ELEMENT_REPRESENTATION('107',(#1143),#1135,(#445,#465,
#561,#589),#67,#1243,#1103);
#1467= VOLUME_3D_ELEMENT_REPRESENTATION('108',(#1143),#1135,(#469,#589,
#445,#473),#67,#1243,#1103);
#1470= VOLUME_3D_ELEMENT_REPRESENTATION('109',(#1143),#1135,(#565,#445,
#589,#569),#67,#1243,#1103);
#1473= VOLUME_3D_ELEMENT_REPRESENTATION('110',(#1143),#1135,(#449,#569,
#473,#445),#67,#1243,#1103);
#1476= VOLUME_3D_ELEMENT_REPRESENTATION('111',(#1143),#1135,(#593,#473,
#569,#589),#67,#1243,#1103);
#1479= VOLUME_3D_ELEMENT_REPRESENTATION('112',(#1143),#1135,(#445,#473,
#589,#569),#67,#1243,#1103);
#1482= VOLUME_3D_ELEMENT_REPRESENTATION('113',(#1143),#1135,(#449,#473,
#569,#453),#67,#1243,#1103);
#1485= VOLUME_3D_ELEMENT_REPRESENTATION('114',(#1143),#1135,(#593,#569,
#473,#597),#67,#1243,#1103);
#1488= VOLUME_3D_ELEMENT_REPRESENTATION('115',(#1143),#1135,(#477,#453,
#597,#473),#67,#1243,#1103);
#1491= VOLUME_3D_ELEMENT_REPRESENTATION('116',(#1143),#1135,(#573,#597,
#453,#569),#67,#1243,#1103);
#1494= VOLUME_3D_ELEMENT_REPRESENTATION('117',(#1143),#1135,(#453,#473,
#569,#597),#67,#1243,#1103);
#1497= VOLUME_3D_ELEMENT_REPRESENTATION('118',(#1143),#1135,(#477,#597,
#453,#329),#67,#1243,#1103);
#1500= VOLUME_3D_ELEMENT_REPRESENTATION('119',(#1143),#1135,(#573,#453,
#597,#353),#67,#1243,#1103);
#1503= VOLUME_3D_ELEMENT_REPRESENTATION('120',(#1143),#1135,(#333,#353,
#329,#453),#67,#1243,#1103);
#1506= VOLUME_3D_ELEMENT_REPRESENTATION('121',(#1143),#1135,(#349,#329,
#353,#597),#67,#1243,#1103);
#1509= VOLUME_3D_ELEMENT_REPRESENTATION('122',(#1143),#1135,(#453,#329,
#597,#353),#67,#1243,#1103);
#1512= VOLUME_3D_ELEMENT_REPRESENTATION('123',(#1143),#1135,(#481,#601,
#457,#485),#67,#1243,#1103);
#1515= VOLUME_3D_ELEMENT_REPRESENTATION('124',(#1143),#1135,(#577,#457,
#601,#581),#67,#1243,#1103);
#1518= VOLUME_3D_ELEMENT_REPRESENTATION('125',(#1143),#1135,(#461,#581,
#485,#457),#67,#1243,#1103);
#1521= VOLUME_3D_ELEMENT_REPRESENTATION('126',(#1143),#1135,(#605,#485,
```

```
#581,#601),#67,#1243,#1103);
#1524= VOLUME_3D_ELEMENT_REPRESENTATION('127',(#1143),#1135,(#457,#485,
#601,#581),#67,#1243,#1103);
#1527= VOLUME_3D_ELEMENT_REPRESENTATION('128',(#1143),#1135,(#461,#485,
#581,#465),#67,#1243,#1103);
#1530= VOLUME_3D_ELEMENT_REPRESENTATION('129',(#1143),#1135,(#605,#581,
#485,#609),#67,#1243,#1103);
#1533= VOLUME_3D_ELEMENT_REPRESENTATION('130',(#1143),#1135,(#489,#465,
#609,#485),#67,#1243,#1103);
#1536= VOLUME_3D_ELEMENT_REPRESENTATION('131',(#1143),#1135,(#585,#609,
#465,#581),#67,#1243,#1103);
#1539= VOLUME_3D_ELEMENT_REPRESENTATION('132',(#1143),#1135,(#465,#485,
#581,#609),#67,#1243,#1103);
#1542= VOLUME_3D_ELEMENT_REPRESENTATION('133',(#1143),#1135,(#489,#609,
#465,#493),#67,#1243,#1103);
#1545= VOLUME_3D_ELEMENT_REPRESENTATION('134',(#1143),#1135,(#585,#465,
#609,#589),#67,#1243,#1103);
#1548= VOLUME_3D_ELEMENT_REPRESENTATION('135',(#1143),#1135,(#469,#589,
#493,#465),#67,#1243,#1103);
#1551= VOLUME_3D_ELEMENT_REPRESENTATION('136',(#1143),#1135,(#613,#493,
#589,#609),#67,#1243,#1103);
#1554= VOLUME_3D_ELEMENT_REPRESENTATION('137',(#1143),#1135,(#465,#493,
#609,#589),#67,#1243,#1103);
#1557= VOLUME_3D_ELEMENT_REPRESENTATION('138',(#1143),#1135,(#469,#493,
#589,#473),#67,#1243,#1103);
#1560= VOLUME_3D_ELEMENT_REPRESENTATION('139',(#1143),#1135,(#613,#589,
#493,#617),#67,#1243,#1103);
#1563= VOLUME_3D_ELEMENT_REPRESENTATION('140',(#1143),#1135,(#497,#473,
#617,#493),#67,#1243,#1103);
#1566= VOLUME_3D_ELEMENT_REPRESENTATION('141',(#1143),#1135,(#593,#617,
#473,#589),#67,#1243,#1103);
#1569= VOLUME_3D_ELEMENT_REPRESENTATION('142',(#1143),#1135,(#473,#493,
#589,#617),#67,#1243,#1103);
#1572= VOLUME_3D_ELEMENT_REPRESENTATION('143',(#1143),#1135,(#497,#617,
#473,#501),#67,#1243,#1103);
#1575= VOLUME_3D_ELEMENT_REPRESENTATION('144',(#1143),#1135,(#593,#473,
#617,#597),#67,#1243,#1103);
#1578= VOLUME_3D_ELEMENT_REPRESENTATION('145',(#1143),#1135,(#477,#597,
#501,#473),#67,#1243,#1103);
#1581= VOLUME_3D_ELEMENT_REPRESENTATION('146',(#1143),#1135,(#621,#501,
#597,#617),#67,#1243,#1103);
#1584= VOLUME_3D_ELEMENT_REPRESENTATION('147',(#1143),#1135,(#473,#501,
#617,#597),#67,#1243,#1103);
#1587= VOLUME_3D_ELEMENT_REPRESENTATION('148',(#1143),#1135,(#477,#501,
#597,#329),#67,#1243,#1103);
#1590= VOLUME_3D_ELEMENT_REPRESENTATION('149',(#1143),#1135,(#621,#597,
#501,#345),#67,#1243,#1103);
#1593= VOLUME_3D_ELEMENT_REPRESENTATION('150',(#1143),#1135,(#325,#329,
#345,#501),#67,#1243,#1103);
#1596= VOLUME_3D_ELEMENT_REPRESENTATION('151',(#1143),#1135,(#349,#345,
#329,#597),#67,#1243,#1103);
#1599= VOLUME_3D_ELEMENT_REPRESENTATION('152',(#1143),#1135,(#329,#501,
#597,#345),#67,#1243,#1103);
#1602= VOLUME_3D_ELEMENT_REPRESENTATION('153',(#1143),#1135,(#529,#649,
#505,#533),#67,#1243,#1103);
#1605= VOLUME_3D_ELEMENT_REPRESENTATION('154',(#1143),#1135,(#625,#505,
#649,#629),#67,#1243,#1103);
#1608= VOLUME_3D_ELEMENT_REPRESENTATION('155',(#1143),#1135,(#509,#629,
#533,#505),#67,#1243,#1103);
#1611= VOLUME_3D_ELEMENT_REPRESENTATION('156',(#1143),#1135,(#653,#533,
#629,#649),#67,#1243,#1103);
#1614= VOLUME_3D_ELEMENT_REPRESENTATION('157',(#1143),#1135,(#505,#533,
#649,#629),#67,#1243,#1103);
#1617= VOLUME_3D_ELEMENT_REPRESENTATION('158',(#1143),#1135,(#509,#533,
#629,#513),#67,#1243,#1103);
#1620= VOLUME_3D_ELEMENT_REPRESENTATION('159',(#1143),#1135,(#653,#629,
#533,#657),#67,#1243,#1103);
#1623= VOLUME_3D_ELEMENT_REPRESENTATION('160',(#1143),#1135,(#537,#513,
#657,#533),#67,#1243,#1103);
#1626= VOLUME_3D_ELEMENT_REPRESENTATION('161',(#1143),#1135,(#633,#657,
#513,#629),#67,#1243,#1103);
```

```
#1629= VOLUME_3D_ELEMENT_REPRESENTATION('162',(#1143),#1135,(#513,#533,
#629,#657),#67,#1243,#1103);
#1632= VOLUME_3D_ELEMENT_REPRESENTATION('163',(#1143),#1135,(#537,#657,
#513,#541),#67,#1243,#1103);
#1635= VOLUME_3D_ELEMENT_REPRESENTATION('164',(#1143),#1135,(#633,#513,
#657,#637),#67,#1243,#1103);
#1638= VOLUME_3D_ELEMENT_REPRESENTATION('165',(#1143),#1135,(#517,#637,
#541,#513),#67,#1243,#1103);
#1641= VOLUME_3D_ELEMENT_REPRESENTATION('166',(#1143),#1135,(#661,#541,
#637,#657),#67,#1243,#1103);
#1644= VOLUME_3D_ELEMENT_REPRESENTATION('167',(#1143),#1135,(#513,#541,
#657,#637),#67,#1243,#1103);
#1647= VOLUME_3D_ELEMENT_REPRESENTATION('168',(#1143),#1135,(#517,#541,
#637,#521),#67,#1243,#1103);
#1650= VOLUME_3D_ELEMENT_REPRESENTATION('169',(#1143),#1135,(#661,#637,
#541,#665),#67,#1243,#1103);
#1653= VOLUME_3D_ELEMENT_REPRESENTATION('170',(#1143),#1135,(#545,#521,
#665,#541),#67,#1243,#1103);
#1656= VOLUME_3D_ELEMENT_REPRESENTATION('171',(#1143),#1135,(#641,#665,
#521,#637),#67,#1243,#1103);
#1659= VOLUME_3D_ELEMENT_REPRESENTATION('172',(#1143),#1135,(#521,#541,
#637,#665),#67,#1243,#1103);
#1662= VOLUME_3D_ELEMENT_REPRESENTATION('173',(#1143),#1135,(#545,#665,
#521,#549),#67,#1243,#1103);
#1665= VOLUME_3D_ELEMENT_REPRESENTATION('174',(#1143),#1135,(#641,#521,
#665,#645),#67,#1243,#1103);
#1668= VOLUME_3D_ELEMENT_REPRESENTATION('175',(#1143),#1135,(#525,#645,
#549,#521),#67,#1243,#1103);
#1671= VOLUME_3D_ELEMENT_REPRESENTATION('176',(#1143),#1135,(#669,#549,
#645,#665),#67,#1243,#1103);
#1674= VOLUME_3D_ELEMENT_REPRESENTATION('177',(#1143),#1135,(#521,#549,
#665,#645),#67,#1243,#1103);
#1677= VOLUME_3D_ELEMENT_REPRESENTATION('178',(#1143),#1135,(#525,#549,
#645,#361),#67,#1243,#1103);
#1680= VOLUME_3D_ELEMENT_REPRESENTATION('179',(#1143),#1135,(#669,#645,
#549,#377),#67,#1243,#1103);
#1683= VOLUME_3D_ELEMENT_REPRESENTATION('180',(#1143),#1135,(#357,#361,
#377,#549),#67,#1243,#1103);
#1686= VOLUME_3D_ELEMENT_REPRESENTATION('181',(#1143),#1135,(#381,#377,
#361,#645),#67,#1243,#1103);
#1689= VOLUME_3D_ELEMENT_REPRESENTATION('182',(#1143),#1135,(#361,#549,
#645,#377),#67,#1243,#1103);
#1692= VOLUME_3D_ELEMENT_REPRESENTATION('183',(#1143),#1135,(#529,#553,
#649,#533),#67,#1243,#1103);
#1695= VOLUME_3D_ELEMENT_REPRESENTATION('184',(#1143),#1135,(#673,#649,
#553,#677),#67,#1243,#1103);
#1698= VOLUME_3D_ELEMENT_REPRESENTATION('185',(#1143),#1135,(#557,#533,
#677,#553),#67,#1243,#1103);
#1701= VOLUME_3D_ELEMENT_REPRESENTATION('186',(#1143),#1135,(#653,#677,
#533,#649),#67,#1243,#1103);
#1704= VOLUME_3D_ELEMENT_REPRESENTATION('187',(#1143),#1135,(#533,#553,
#649,#677),#67,#1243,#1103);
#1707= VOLUME_3D_ELEMENT_REPRESENTATION('188',(#1143),#1135,(#557,#677,
#533,#561),#67,#1243,#1103);
#1710= VOLUME_3D_ELEMENT_REPRESENTATION('189',(#1143),#1135,(#653,#533,
#677,#657),#67,#1243,#1103);
#1713= VOLUME_3D_ELEMENT_REPRESENTATION('190',(#1143),#1135,(#537,#657,
#561,#533),#67,#1243,#1103);
#1716= VOLUME_3D_ELEMENT_REPRESENTATION('191',(#1143),#1135,(#681,#561,
#657,#677),#67,#1243,#1103);
#1719= VOLUME_3D_ELEMENT_REPRESENTATION('192',(#1143),#1135,(#533,#561,
#677,#657),#67,#1243,#1103);
#1722= VOLUME_3D_ELEMENT_REPRESENTATION('193',(#1143),#1135,(#537,#561,
#657,#541),#67,#1243,#1103);
#1725= VOLUME_3D_ELEMENT_REPRESENTATION('194',(#1143),#1135,(#681,#657,
#561,#685),#67,#1243,#1103);
#1728= VOLUME_3D_ELEMENT_REPRESENTATION('195',(#1143),#1135,(#565,#541,
#685,#561),#67,#1243,#1103);
#1731= VOLUME_3D_ELEMENT_REPRESENTATION('196',(#1143),#1135,(#661,#685,
#541,#657),#67,#1243,#1103);
#1734= VOLUME_3D_ELEMENT_REPRESENTATION('197',(#1143),#1135,(#541,#561,
```

```
#657,#685),#67,#1243,#1103);
#1737= VOLUME_3D_ELEMENT_REPRESENTATION('198',(#1143),#1135,(#565,#685,
#541,#569),#67,#1243,#1103);
#1740= VOLUME_3D_ELEMENT_REPRESENTATION('199',(#1143),#1135,(#661,#541,
#685,#665),#67,#1243,#1103);
#1743= VOLUME_3D_ELEMENT_REPRESENTATION('200',(#1143),#1135,(#545,#665,
#569,#541),#67,#1243,#1103);
#1746= VOLUME_3D_ELEMENT_REPRESENTATION('201',(#1143),#1135,(#689,#569,
#665,#685),#67,#1243,#1103);
#1749= VOLUME_3D_ELEMENT_REPRESENTATION('202',(#1143),#1135,(#541,#569,
#685,#665),#67,#1243,#1103);
#1752= VOLUME_3D_ELEMENT_REPRESENTATION('203',(#1143),#1135,(#545,#569,
#665,#549),#67,#1243,#1103);
#1755= VOLUME_3D_ELEMENT_REPRESENTATION('204',(#1143),#1135,(#689,#665,
#569,#693),#67,#1243,#1103);
#1758= VOLUME_3D_ELEMENT_REPRESENTATION('205',(#1143),#1135,(#573,#549,
#693,#569),#67,#1243,#1103);
#1761= VOLUME_3D_ELEMENT_REPRESENTATION('206',(#1143),#1135,(#669,#693,
#549,#665),#67,#1243,#1103);
#1764= VOLUME_3D_ELEMENT_REPRESENTATION('207',(#1143),#1135,(#549,#569,
#665,#693),#67,#1243,#1103);
#1767= VOLUME_3D_ELEMENT_REPRESENTATION('208',(#1143),#1135,(#573,#693,
#549,#353),#67,#1243,#1103);
#1770= VOLUME_3D_ELEMENT_REPRESENTATION('209',(#1143),#1135,(#669,#549,
#693,#377),#67,#1243,#1103);
#1773= VOLUME_3D_ELEMENT_REPRESENTATION('210',(#1143),#1135,(#357,#377,
#353,#549),#67,#1243,#1103);
#1776= VOLUME_3D_ELEMENT_REPRESENTATION('211',(#1143),#1135,(#373,#353,
#377,#693),#67,#1243,#1103);
#1779= VOLUME_3D_ELEMENT_REPRESENTATION('212',(#1143),#1135,(#549,#353,
#693,#377),#67,#1243,#1103);
#1782= VOLUME_3D_ELEMENT_REPRESENTATION('213',(#1143),#1135,(#577,#697,
#553,#581),#67,#1243,#1103);
#1785= VOLUME_3D_ELEMENT_REPRESENTATION('214',(#1143),#1135,(#673,#553,
#697,#677),#67,#1243,#1103);
#1788= VOLUME_3D_ELEMENT_REPRESENTATION('215',(#1143),#1135,(#557,#677,
#581,#553),#67,#1243,#1103);
#1791= VOLUME_3D_ELEMENT_REPRESENTATION('216',(#1143),#1135,(#701,#581,
#677,#697),#67,#1243,#1103);
#1794= VOLUME_3D_ELEMENT_REPRESENTATION('217',(#1143),#1135,(#553,#581,
#697,#677),#67,#1243,#1103);
#1797= VOLUME_3D_ELEMENT_REPRESENTATION('218',(#1143),#1135,(#557,#581,
#677,#561),#67,#1243,#1103);
#1800= VOLUME_3D_ELEMENT_REPRESENTATION('219',(#1143),#1135,(#701,#677,
#581,#705),#67,#1243,#1103);
#1803= VOLUME_3D_ELEMENT_REPRESENTATION('220',(#1143),#1135,(#585,#561,
#705,#581),#67,#1243,#1103);
#1806= VOLUME_3D_ELEMENT_REPRESENTATION('221',(#1143),#1135,(#681,#705,
#561,#677),#67,#1243,#1103);
#1809= VOLUME_3D_ELEMENT_REPRESENTATION('222',(#1143),#1135,(#561,#581,
#677,#705),#67,#1243,#1103);
#1812= VOLUME_3D_ELEMENT_REPRESENTATION('223',(#1143),#1135,(#585,#705,
#561,#589),#67,#1243,#1103);
#1815= VOLUME_3D_ELEMENT_REPRESENTATION('224',(#1143),#1135,(#681,#561,
#705,#685),#67,#1243,#1103);
#1818= VOLUME_3D_ELEMENT_REPRESENTATION('225',(#1143),#1135,(#565,#685,
#589,#561),#67,#1243,#1103);
#1821= VOLUME_3D_ELEMENT_REPRESENTATION('226',(#1143),#1135,(#709,#589,
#685,#705),#67,#1243,#1103);
#1824= VOLUME_3D_ELEMENT_REPRESENTATION('227',(#1143),#1135,(#561,#589,
#705,#685),#67,#1243,#1103);
#1827= VOLUME_3D_ELEMENT_REPRESENTATION('228',(#1143),#1135,(#565,#589,
#685,#569),#67,#1243,#1103);
#1830= VOLUME_3D_ELEMENT_REPRESENTATION('229',(#1143),#1135,(#709,#685,
#589,#713),#67,#1243,#1103);
#1833= VOLUME_3D_ELEMENT_REPRESENTATION('230',(#1143),#1135,(#593,#569,
#713,#589),#67,#1243,#1103);
#1836= VOLUME_3D_ELEMENT_REPRESENTATION('231',(#1143),#1135,(#689,#713,
#569,#685),#67,#1243,#1103);
#1839= VOLUME_3D_ELEMENT_REPRESENTATION('232',(#1143),#1135,(#569,#589,
#685,#713),#67,#1243,#1103);
```

```
#1842= VOLUME_3D_ELEMENT_REPRESENTATION('233',(#1143),#1135,(#593,#713,
#569,#597),#67,#1243,#1103);
#1845= VOLUME_3D_ELEMENT_REPRESENTATION('234',(#1143),#1135,(#689,#569,
#713,#693),#67,#1243,#1103);
#1848= VOLUME_3D_ELEMENT_REPRESENTATION('235',(#1143),#1135,(#573,#693,
#597,#569),#67,#1243,#1103);
#1851= VOLUME_3D_ELEMENT_REPRESENTATION('236',(#1143),#1135,(#717,#597,
#693,#713),#67,#1243,#1103);
#1854= VOLUME_3D_ELEMENT_REPRESENTATION('237',(#1143),#1135,(#569,#597,
#713,#693),#67,#1243,#1103);
#1857= VOLUME_3D_ELEMENT_REPRESENTATION('238',(#1143),#1135,(#573,#597,
#693,#353),#67,#1243,#1103);
#1860= VOLUME_3D_ELEMENT_REPRESENTATION('239',(#1143),#1135,(#717,#693,
#597,#369),#67,#1243,#1103);
#1863= VOLUME_3D_ELEMENT_REPRESENTATION('240',(#1143),#1135,(#349,#353,
#369,#597),#67,#1243,#1103);
#1866= VOLUME_3D_ELEMENT_REPRESENTATION('241',(#1143),#1135,(#373,#369,
#353,#693),#67,#1243,#1103);
#1869= VOLUME_3D_ELEMENT_REPRESENTATION('242',(#1143),#1135,(#353,#597,
#693,#369),#67,#1243,#1103);
#1872= VOLUME_3D_ELEMENT_REPRESENTATION('243',(#1143),#1135,(#577,#601,
#697,#581),#67,#1243,#1103);
#1875= VOLUME_3D_ELEMENT_REPRESENTATION('244',(#1143),#1135,(#721,#697,
#601,#725),#67,#1243,#1103);
#1878= VOLUME_3D_ELEMENT_REPRESENTATION('245',(#1143),#1135,(#605,#581,
#725,#601),#67,#1243,#1103);
#1881= VOLUME_3D_ELEMENT_REPRESENTATION('246',(#1143),#1135,(#701,#725,
#581,#697),#67,#1243,#1103);
#1884= VOLUME_3D_ELEMENT_REPRESENTATION('247',(#1143),#1135,(#581,#601,
#697,#725),#67,#1243,#1103);
#1887= VOLUME_3D_ELEMENT_REPRESENTATION('248',(#1143),#1135,(#605,#725,
#581,#609),#67,#1243,#1103);
#1890= VOLUME_3D_ELEMENT_REPRESENTATION('249',(#1143),#1135,(#701,#581,
#725,#705),#67,#1243,#1103);
#1893= VOLUME_3D_ELEMENT_REPRESENTATION('250',(#1143),#1135,(#585,#705,
#609,#581),#67,#1243,#1103);
#1896= VOLUME_3D_ELEMENT_REPRESENTATION('251',(#1143),#1135,(#729,#609,
#705,#725),#67,#1243,#1103);
#1899= VOLUME_3D_ELEMENT_REPRESENTATION('252',(#1143),#1135,(#581,#609,
#725,#705),#67,#1243,#1103);
#1902= VOLUME_3D_ELEMENT_REPRESENTATION('253',(#1143),#1135,(#585,#609,
#705,#589),#67,#1243,#1103);
#1905= VOLUME_3D_ELEMENT_REPRESENTATION('254',(#1143),#1135,(#729,#705,
#609,#733),#67,#1243,#1103);
#1908= VOLUME_3D_ELEMENT_REPRESENTATION('255',(#1143),#1135,(#613,#589,
#733,#609),#67,#1243,#1103);
#1911= VOLUME_3D_ELEMENT_REPRESENTATION('256',(#1143),#1135,(#709,#733,
#589,#705),#67,#1243,#1103);
#1914= VOLUME_3D_ELEMENT_REPRESENTATION('257',(#1143),#1135,(#589,#609,
#705,#733),#67,#1243,#1103);
#1917= VOLUME_3D_ELEMENT_REPRESENTATION('258',(#1143),#1135,(#613,#733,
#589,#617),#67,#1243,#1103);
#1920= VOLUME_3D_ELEMENT_REPRESENTATION('259',(#1143),#1135,(#709,#589,
#733,#713),#67,#1243,#1103);
#1923= VOLUME_3D_ELEMENT_REPRESENTATION('260',(#1143),#1135,(#593,#713,
#617,#589),#67,#1243,#1103);
#1926= VOLUME_3D_ELEMENT_REPRESENTATION('261',(#1143),#1135,(#737,#617,
#713,#733),#67,#1243,#1103);
#1929= VOLUME_3D_ELEMENT_REPRESENTATION('262',(#1143),#1135,(#589,#617,
#733,#713),#67,#1243,#1103);
#1932= VOLUME_3D_ELEMENT_REPRESENTATION('263',(#1143),#1135,(#593,#617,
#713,#597),#67,#1243,#1103);
#1935= VOLUME_3D_ELEMENT_REPRESENTATION('264',(#1143),#1135,(#737,#713,
#617,#741),#67,#1243,#1103);
#1938= VOLUME_3D_ELEMENT_REPRESENTATION('265',(#1143),#1135,(#621,#597,
#741,#617),#67,#1243,#1103);
#1941= VOLUME_3D_ELEMENT_REPRESENTATION('266',(#1143),#1135,(#717,#741,
#597,#713),#67,#1243,#1103);
#1944= VOLUME_3D_ELEMENT_REPRESENTATION('267',(#1143),#1135,(#597,#617,
#713,#741),#67,#1243,#1103);
#1947= VOLUME_3D_ELEMENT_REPRESENTATION('268',(#1143),#1135,(#621,#741,
```

```
#597,#345),#67,#1243,#1103);
#1950= VOLUME_3D_ELEMENT_REPRESENTATION('269',(#1143),#1135,(#717,#597,
#741,#369),#67,#1243,#1103);
#1953= VOLUME_3D_ELEMENT_REPRESENTATION('270',(#1143),#1135,(#349,#369,
#345,#597),#67,#1243,#1103);
#1956= VOLUME_3D_ELEMENT_REPRESENTATION('271',(#1143),#1135,(#365,#345,
#369,#741),#67,#1243,#1103);
#1959= VOLUME_3D_ELEMENT_REPRESENTATION('272',(#1143),#1135,(#597,#345,
#741,#369),#67,#1243,#1103);
#1962= VOLUME_3D_ELEMENT_REPRESENTATION('273',(#1143),#1135,(#773,#769,
#745,#893,#889,#865),#67,#1965,#1103);
#1965= VOLUME_3D_ELEMENT_DESCRIPTOR(.LINEAR_ORDER.,'LINEAR_WEDGE.CPENTA'
,(ENUMERATED_VOLUME_ELEMENT_PURPOSE(.STRESS_DISPLACEMENT.)),.WEDGE.);
#1967= VOLUME_3D_ELEMENT_REPRESENTATION('274',(#1143),#1135,(#745,#749,
#773,#865,#869,#893),#67,#1965,#1103);
#1970= VOLUME_3D_ELEMENT_REPRESENTATION('275',(#1143),#1135,(#749,#753,
#773,#869,#873,#893),#67,#1965,#1103);
#1973= VOLUME_3D_ELEMENT_REPRESENTATION('276',(#1143),#1135,(#777,#773,
#753,#897,#893,#873),#67,#1965,#1103);
#1976= VOLUME_3D_ELEMENT_REPRESENTATION('277',(#1143),#1135,(#781,#777,
#753,#901,#897,#873),#67,#1965,#1103);
#1979= VOLUME_3D_ELEMENT_REPRESENTATION('278',(#1143),#1135,(#753,#757,
#781,#873,#877,#901),#67,#1965,#1103);
#1982= VOLUME_3D_ELEMENT_REPRESENTATION('279',(#1143),#1135,(#757,#761,
#781,#877,#881,#901),#67,#1965,#1103);
#1985= VOLUME_3D_ELEMENT_REPRESENTATION('280',(#1143),#1135,(#785,#781,
#761,#905,#901,#881),#67,#1965,#1103);
#1988= VOLUME_3D_ELEMENT_REPRESENTATION('281',(#1143),#1135,(#789,#785,
#761,#909,#905,#881),#67,#1965,#1103);
#1991= VOLUME_3D_ELEMENT_REPRESENTATION('282',(#1143),#1135,(#761,#765,
#789,#881,#885,#909),#67,#1965,#1103);
#1994= VOLUME_3D_ELEMENT_REPRESENTATION('283',(#1143),#1135,(#765,#385,
#789,#885,#505,#909),#67,#1965,#1103);
#1997= VOLUME_3D_ELEMENT_REPRESENTATION('284',(#1143),#1135,(#409,#789,
#385,#529,#909,#505),#67,#1965,#1103);
#2000= VOLUME_3D_ELEMENT_REPRESENTATION('285',(#1143),#1135,(#769,#773,
#793,#889,#893,#913),#67,#1965,#1103);
#2003= VOLUME_3D_ELEMENT_REPRESENTATION('286',(#1143),#1135,(#797,#793,
#773,#917,#913,#893),#67,#1965,#1103);
#2006= VOLUME_3D_ELEMENT_REPRESENTATION('287',(#1143),#1135,(#801,#797,
#773,#921,#917,#893),#67,#1965,#1103);
#2009= VOLUME_3D_ELEMENT_REPRESENTATION('288',(#1143),#1135,(#773,#777,
#801,#893,#897,#921),#67,#1965,#1103);
#2012= VOLUME_3D_ELEMENT_REPRESENTATION('289',(#1143),#1135,(#777,#781,
#801,#897,#901,#921),#67,#1965,#1103);
#2015= VOLUME_3D_ELEMENT_REPRESENTATION('290',(#1143),#1135,(#805,#801,
#781,#925,#921,#901),#67,#1965,#1103);
#2018= VOLUME_3D_ELEMENT_REPRESENTATION('291',(#1143),#1135,(#809,#805,
#781,#929,#925,#901),#67,#1965,#1103);
#2021= VOLUME_3D_ELEMENT_REPRESENTATION('292',(#1143),#1135,(#781,#785,
#809,#901,#905,#929),#67,#1965,#1103);
#2024= VOLUME_3D_ELEMENT_REPRESENTATION('293',(#1143),#1135,(#785,#789,
#809,#905,#909,#929),#67,#1965,#1103);
#2027= VOLUME_3D_ELEMENT_REPRESENTATION('294',(#1143),#1135,(#813,#809,
#789,#933,#929,#909),#67,#1965,#1103);
#2030= VOLUME_3D_ELEMENT_REPRESENTATION('295',(#1143),#1135,(#433,#813,
#789,#553,#933,#909),#67,#1965,#1103);
#2033= VOLUME_3D_ELEMENT_REPRESENTATION('296',(#1143),#1135,(#789,#409,
#433,#909,#529,#553),#67,#1965,#1103);
#2036= VOLUME_3D_ELEMENT_REPRESENTATION('297',(#1143),#1135,(#821,#817,
#793,#941,#937,#913),#67,#1965,#1103);
#2039= VOLUME_3D_ELEMENT_REPRESENTATION('298',(#1143),#1135,(#793,#797,
#821,#913,#917,#941),#67,#1965,#1103);
#2042= VOLUME_3D_ELEMENT_REPRESENTATION('299',(#1143),#1135,(#797,#801,
#821,#917,#921,#941),#67,#1965,#1103);
#2045= VOLUME_3D_ELEMENT_REPRESENTATION('300',(#1143),#1135,(#825,#821,
#801,#945,#941,#921),#67,#1965,#1103);
#2048= VOLUME_3D_ELEMENT_REPRESENTATION('301',(#1143),#1135,(#829,#825,
#801,#949,#945,#921),#67,#1965,#1103);
#2051= VOLUME_3D_ELEMENT_REPRESENTATION('302',(#1143),#1135,(#801,#805,
#829,#921,#925,#949),#67,#1965,#1103);
```

```
#2054= VOLUME_3D_ELEMENT_REPRESENTATION('303',(#1143),#1135,(#805,#809,
#829,#925,#929,#949),#67,#1965,#1103);
#2057= VOLUME_3D_ELEMENT_REPRESENTATION('304',(#1143),#1135,(#833,#829,
#809,#953,#949,#929),#67,#1965,#1103);
#2060= VOLUME_3D_ELEMENT_REPRESENTATION('305',(#1143),#1135,(#837,#833,
#809,#957,#953,#929),#67,#1965,#1103);
#2063= VOLUME_3D_ELEMENT_REPRESENTATION('306',(#1143),#1135,(#809,#813,
#837,#929,#933,#957),#67,#1965,#1103);
#2066= VOLUME_3D_ELEMENT_REPRESENTATION('307',(#1143),#1135,(#813,#433,
#837,#933,#553,#957),#67,#1965,#1103);
#2069= VOLUME_3D_ELEMENT_REPRESENTATION('308',(#1143),#1135,(#457,#837,
#433,#577,#957,#553),#67,#1965,#1103);
#2072= VOLUME_3D_ELEMENT_REPRESENTATION('309',(#1143),#1135,(#817,#821,
#841,#937,#941,#961),#67,#1965,#1103);
#2075= VOLUME_3D_ELEMENT_REPRESENTATION('310',(#1143),#1135,(#845,#841,
#821,#965,#961,#941),#67,#1965,#1103);
#2078= VOLUME_3D_ELEMENT_REPRESENTATION('311',(#1143),#1135,(#849,#845,
#821,#969,#965,#941),#67,#1965,#1103);
#2081= VOLUME_3D_ELEMENT_REPRESENTATION('312',(#1143),#1135,(#821,#825,
#849,#941,#945,#969),#67,#1965,#1103);
#2084= VOLUME_3D_ELEMENT_REPRESENTATION('313',(#1143),#1135,(#825,#829,
#849,#945,#949,#969),#67,#1965,#1103);
#2087= VOLUME_3D_ELEMENT_REPRESENTATION('314',(#1143),#1135,(#853,#849,
#829,#973,#969,#949),#67,#1965,#1103);
#2090= VOLUME_3D_ELEMENT_REPRESENTATION('315',(#1143),#1135,(#857,#853,
#829,#977,#973,#949),#67,#1965,#1103);
#2093= VOLUME_3D_ELEMENT_REPRESENTATION('316',(#1143),#1135,(#829,#833,
#857,#949,#953,#977),#67,#1965,#1103);
#2096= VOLUME_3D_ELEMENT_REPRESENTATION('317',(#1143),#1135,(#833,#837,
#857,#953,#957,#977),#67,#1965,#1103);
#2099= VOLUME_3D_ELEMENT_REPRESENTATION('318',(#1143),#1135,(#861,#857,
#837,#981,#977,#957),#67,#1965,#1103);
#2102= VOLUME_3D_ELEMENT_REPRESENTATION('319',(#1143),#1135,(#481,#861,
#837,#601,#981,#957),#67,#1965,#1103);
#2105= VOLUME_3D_ELEMENT_REPRESENTATION('320',(#1143),#1135,(#837,#457,
#481,#957,#577,#601),#67,#1965,#1103);
#2108= VOLUME_3D_ELEMENT_REPRESENTATION('321',(#1143),#1135,(#893,#889,
#865,#1013,#1009,#985),#67,#1965,#1103);
#2111= VOLUME_3D_ELEMENT_REPRESENTATION('322',(#1143),#1135,(#865,#869,
#893,#985,#989,#1013),#67,#1965,#1103);
#2114= VOLUME_3D_ELEMENT_REPRESENTATION('323',(#1143),#1135,(#869,#873,
#893,#989,#993,#1013),#67,#1965,#1103);
#2117= VOLUME_3D_ELEMENT_REPRESENTATION('324',(#1143),#1135,(#897,#893,
#873,#1017,#1013,#993),#67,#1965,#1103);
#2120= VOLUME_3D_ELEMENT_REPRESENTATION('325',(#1143),#1135,(#901,#897,
#873,#1021,#1017,#993),#67,#1965,#1103);
#2123= VOLUME_3D_ELEMENT_REPRESENTATION('326',(#1143),#1135,(#873,#877,
#901,#993,#997,#1021),#67,#1965,#1103);
#2126= VOLUME_3D_ELEMENT_REPRESENTATION('327',(#1143),#1135,(#877,#881,
#901,#997,#1001,#1021),#67,#1965,#1103);
#2129= VOLUME_3D_ELEMENT_REPRESENTATION('328',(#1143),#1135,(#905,#901,
#881,#1025,#1021,#1001),#67,#1965,#1103);
#2132= VOLUME_3D_ELEMENT_REPRESENTATION('329',(#1143),#1135,(#909,#905,
#881,#1029,#1025,#1001),#67,#1965,#1103);
#2135= VOLUME_3D_ELEMENT_REPRESENTATION('330',(#1143),#1135,(#881,#885,
#909,#1001,#1005,#1029),#67,#1965,#1103);
#2138= VOLUME_3D_ELEMENT_REPRESENTATION('331',(#1143),#1135,(#885,#505,
#909,#1005,#625,#1029),#67,#1965,#1103);
#2141= VOLUME_3D_ELEMENT_REPRESENTATION('332',(#1143),#1135,(#529,#909,
#505,#649,#1029,#625),#67,#1965,#1103);
#2144= VOLUME_3D_ELEMENT_REPRESENTATION('333',(#1143),#1135,(#889,#893,
#913,#1009,#1013,#1033),#67,#1965,#1103);
#2147= VOLUME_3D_ELEMENT_REPRESENTATION('334',(#1143),#1135,(#917,#913,
#893,#1037,#1033,#1013),#67,#1965,#1103);
#2150= VOLUME_3D_ELEMENT_REPRESENTATION('335',(#1143),#1135,(#921,#917,
#893,#1041,#1037,#1013),#67,#1965,#1103);
#2153= VOLUME_3D_ELEMENT_REPRESENTATION('336',(#1143),#1135,(#893,#897,
#921,#1013,#1017,#1041),#67,#1965,#1103);
#2156= VOLUME_3D_ELEMENT_REPRESENTATION('337',(#1143),#1135,(#897,#901,
#921,#1017,#1021,#1041),#67,#1965,#1103);
#2159= VOLUME_3D_ELEMENT_REPRESENTATION('338',(#1143),#1135,(#925,#921,
```

```
#901,#1045,#1041,#1021),#67,#1965,#1103);
#2162= VOLUME_3D_ELEMENT_REPRESENTATION('339',(#1143),#1135,(#929,#925,
#901,#1049,#1045,#1021),#67,#1965,#1103);
#2165= VOLUME_3D_ELEMENT_REPRESENTATION('340',(#1143),#1135,(#901,#905,
#929,#1021,#1025,#1049),#67,#1965,#1103);
#2168= VOLUME_3D_ELEMENT_REPRESENTATION('341',(#1143),#1135,(#905,#909,
#929,#1025,#1029,#1049),#67,#1965,#1103);
#2171= VOLUME_3D_ELEMENT_REPRESENTATION('342',(#1143),#1135,(#933,#929,
#909,#1053,#1049,#1029),#67,#1965,#1103);
#2174= VOLUME_3D_ELEMENT_REPRESENTATION('343',(#1143),#1135,(#553,#933,
#909,#673,#1053,#1029),#67,#1965,#1103);
#2177= VOLUME_3D_ELEMENT_REPRESENTATION('344',(#1143),#1135,(#909,#529,
#553,#1029,#649,#673),#67,#1965,#1103);
#2180= VOLUME_3D_ELEMENT_REPRESENTATION('345',(#1143),#1135,(#941,#937,
#913,#1061,#1057,#1033),#67,#1965,#1103);
#2183= VOLUME_3D_ELEMENT_REPRESENTATION('346',(#1143),#1135,(#913,#917,
#941,#1033,#1037,#1061),#67,#1965,#1103);
#2186= VOLUME_3D_ELEMENT_REPRESENTATION('347',(#1143),#1135,(#917,#921,
#941,#1037,#1041,#1061),#67,#1965,#1103);
#2189= VOLUME_3D_ELEMENT_REPRESENTATION('348',(#1143),#1135,(#945,#941,
#921,#1065,#1061,#1041),#67,#1965,#1103);
#2192= VOLUME_3D_ELEMENT_REPRESENTATION('349',(#1143),#1135,(#949,#945,
#921,#1069,#1065,#1041),#67,#1965,#1103);
#2195= VOLUME_3D_ELEMENT_REPRESENTATION('350',(#1143),#1135,(#921,#925,
#949,#1041,#1045,#1069),#67,#1965,#1103);
#2198= VOLUME_3D_ELEMENT_REPRESENTATION('351',(#1143),#1135,(#925,#929,
#949,#1045,#1049,#1069),#67,#1965,#1103);
#2201= VOLUME_3D_ELEMENT_REPRESENTATION('352',(#1143),#1135,(#953,#949,
#929,#1073,#1069,#1049),#67,#1965,#1103);
#2204= VOLUME_3D_ELEMENT_REPRESENTATION('353',(#1143),#1135,(#957,#953,
#929,#1077,#1073,#1049),#67,#1965,#1103);
#2207= VOLUME_3D_ELEMENT_REPRESENTATION('354',(#1143),#1135,(#929,#933,
#957,#1049,#1053,#1077),#67,#1965,#1103);
#2210= VOLUME_3D_ELEMENT_REPRESENTATION('355',(#1143),#1135,(#933,#553,
#957,#1053,#673,#1077),#67,#1965,#1103);
#2213= VOLUME_3D_ELEMENT_REPRESENTATION('356',(#1143),#1135,(#577,#957,
#553,#697,#1077,#673),#67,#1965,#1103);
#2216= VOLUME_3D_ELEMENT_REPRESENTATION('357',(#1143),#1135,(#937,#941,
#961,#1057,#1061,#1081),#67,#1965,#1103);
#2219= VOLUME_3D_ELEMENT_REPRESENTATION('358',(#1143),#1135,(#965,#961,
#941,#1085,#1081,#1061),#67,#1965,#1103);
#2222= VOLUME_3D_ELEMENT_REPRESENTATION('359',(#1143),#1135,(#969,#965,
#941,#1089,#1085,#1061),#67,#1965,#1103);
#2225= VOLUME_3D_ELEMENT_REPRESENTATION('360',(#1143),#1135,(#941,#945,
#969,#1061,#1065,#1089),#67,#1965,#1103);
#2228= VOLUME_3D_ELEMENT_REPRESENTATION('361',(#1143),#1135,(#945,#949,
#969,#1065,#1069,#1089),#67,#1965,#1103);
#2231= VOLUME_3D_ELEMENT_REPRESENTATION('362',(#1143),#1135,(#973,#969,
#949,#1093,#1089,#1069),#67,#1965,#1103);
#2234= VOLUME_3D_ELEMENT_REPRESENTATION('363',(#1143),#1135,(#977,#973,
#949,#1097,#1093,#1069),#67,#1965,#1103);
#2237= VOLUME_3D_ELEMENT_REPRESENTATION('364',(#1143),#1135,(#949,#953,
#977,#1069,#1073,#1097),#67,#1965,#1103);
#2240= VOLUME_3D_ELEMENT_REPRESENTATION('365',(#1143),#1135,(#953,#957,
#977,#1073,#1077,#1097),#67,#1965,#1103);
#2243= VOLUME_3D_ELEMENT_REPRESENTATION('366',(#1143),#1135,(#981,#977,
#957,#1101,#1097,#1077),#67,#1965,#1103);
#2246= VOLUME_3D_ELEMENT_REPRESENTATION('367',(#1143),#1135,(#601,#981,
#957,#721,#1101,#1077),#67,#1965,#1103);
#2249= VOLUME_3D_ELEMENT_REPRESENTATION('368',(#1143),#1135,(#957,#577,
#601,#1077,#697,#721),#67,#1965,#1103);
#2252= CONTROL(#67,'Control.0','FemConvert',
'Nastran job EAS test case ATS4m4',('<SOL>101</SOL>','TIME 600','CEND',
'<GLOBALCASE>0</GLOBALCASE>','<SUBCASE>1</SUBCASE>',
'<SUBCASE>2</SUBCASE>','<SUBCASE>3</SUBCASE>','ENDDATA'),('NASTRAN'));
#2255= SPECIFIED_STATE('STEP DEFAULT','default_initial_state');
#2256= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#2252,'STATIC STEP 1',1,#2255
,'',#2396);
#2257= SPECIFIED_STATE('Step 1 Base Specified State',
'Relating Specified State');
#2258= SPECIFIED_STATE('SPCCASE_11_21_1','Aggregator SPC 11 Step 1');
```

```
#2259= SPECIFIED_STATE('SPCVALSTATE_100_21','');
#2260= STATE_RELATIONSHIP(
'SPCVALSTATE_100_21 is related to SPCCASE_11_21_1','',#2258,#2259);
#2261= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2259,#2264,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2262= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#2264= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.0',(#2256),#113,#35,(
#2266,#2269,#2270),'');
#2266= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.X_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#2269= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.Y_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#2270= FREEDOM_AND_COEFFICIENT(ENUMERATED_DEGREE_OF_FREEDOM(
.Z_TRANSLATION.),CONTEXT_DEPENDENT_MEASURE(1.));
#2271= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2259,#2274,#2272,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2272= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#2274= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.1',(#2256),#105,#35,(
#2270),'');
#2278= SPECIFIED_STATE('SPCVALSTATE_101_21','');
#2279= STATE_RELATIONSHIP(
'SPCVALSTATE_101_21 is related to SPCCASE_11_21_1','',#2258,#2278);
#2280= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2283,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2281= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.)));
#2283= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.2',(#2256),#80,#35,(
#2266),'');
#2287= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2288,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2288= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.3',(#2256),#89,#35,(
#2266),'');
#2292= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2293,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2293= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.4',(#2256),#93,#35,(
#2266),'');
#2297= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2298,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2298= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.5',(#2256),#97,#35,(
#2266),'');
#2302= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2303,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2303= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.6',(#2256),#101,#35,(
#2266),'');
#2307= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2308,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2308= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.7',(#2256),#105,#35,(
#2266),'');
#2312= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2313,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2313= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.8',(#2256),#109,#35,(
#2266),'');
#2317= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2318,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2318= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.9',(#2256),#117,#35,(
#2266),'');
#2322= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2323,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2323= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.10',(#2256),#121,#35,(
#2266),'');
#2327= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2328,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2328= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.11',(#2256),#125,#35,(
#2266),'');
#2332= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2333,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2333= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.12',(#2256),#129,#35,(
#2266),'');
#2337= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2338,#2281,(
```

```
CONTEXT_DEPENDENT_MEASURE(0.)));
#2338= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.13',(#2256),#133,#35,(
#2266),'');
#2342= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2343,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2343= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.14',(#2256),#137,#35,(
#2266),'');
#2347= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2278,#2348,#2281,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2348= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_101.15',(#2256),#141,#35,(
#2266),'');
#2352= STATE_RELATIONSHIP(
'SPCCASE_11_21_1 is related to Step 1 Base Specified State','',#2257,
#2258);
#2353= LINEARLY_SUPERIMPOSED_STATE('LOADSTATECOMBINATION_21',
'Overall Factor Combined State');
#2355= STATE_COMPONENT('OverallComp','',#2353,1.);
#2357= SPECIFIED_STATE('LOADSTATECORE_200','Core Loads Specified State'
);
#2358= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#745,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-31.25),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2359= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.)));
#2362= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#841,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-31.25),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2364= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#985,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-31.25),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2366= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#1081,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-31.25),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2368= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#769,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2370= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#793,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2372= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#817,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2374= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#865,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2376= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#961,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2378= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#1009,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2380= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#1033,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2382= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#1057,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2384= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#889,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2386= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#913,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2388= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#937,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2390= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_200',
'Item Factor State');
#2391= STATE_RELATIONSHIP('LOADSTATEITEM_200 is related to OverallComp',
'',#2355,#2390);
```

```
#2392= STATE_COMPONENT('ItemComp_21_200','',#2390,1.);
#2394= STATE_RELATIONSHIP(
'LOADSTATECORE_200 is related to ItemComp_21_200','',#2392,#2357);
#2395= STATE_RELATIONSHIP(
'LOADSTATECOMBINATION_21 is related to Step 1 Base Specified State','',
#2257,#2353);
#2396= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 1','',
#2257);
#2397= OUTPUT_REQUEST_STATE('Step 1 Output Request State','',(#2256));
#2399= NODAL_FREEDOM_VALUES(#2397,#2403,#35,#2400,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#2400= FREEDOMS_LIST((ENUMERATED_DEGREE_OF_FREEDOM(.X_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Y_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.Z_TRANSLATION.),
ENUMERATED_DEGREE_OF_FREEDOM(.X_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(
.Y_ROTATION.),ENUMERATED_DEGREE_OF_FREEDOM(.Z_ROTATION.)));
#2403= NODE_GROUP('','ALL',#67,(#80,#89,#93,#97,#101,#105,#109,#113,#117
,#121,#125,#129,#133,#137,#141,#145,#149,#153,#157,#161,#165,#169,#173,
#177,#181,#185,#189,#193,#197,#201,#205,#209,#213,#217,#221,#225,#229,
#233,#237,#241,#245,#249,#253,#257,#261,#265,#269,#273,#277,#281,#285,
#289,#293,#297,#301,#305,#309,#313,#317,#321,#325,#329,#333,#337,#341,
#345,#349,#353,#357,#361,#365,#369,#373,#377,#381,#385,#389,#393,#397,
#401,#405,#409,#413,#417,#421,#425,#429,#433,#437,#441,#445,#449,#453,
#457,#461,#465,#469,#473,#477,#481,#485,#489,#493,#497,#501,#505,#509,
#513,#517,#521,#525,#529,#533,#537,#541,#545,#549,#553,#557,#561,#565,
#569,#573,#577,#581,#585,#589,#593,#597,#601,#605,#609,#613,#617,#621,
#625,#629,#633,#637,#641,#645,#649,#653,#657,#661,#665,#669,#673,#677,
#681,#685,#689,#693,#697,#701,#705,#709,#713,#717,#721,#725,#729,#733,
#737,#741,#745,#749,#753,#757,#761,#765,#769,#773,#777,#781,#785,#789,
#793,#797,#801,#805,#809,#813,#817,#821,#825,#829,#833,#837,#841,#845,
#849,#853,#857,#861,#865,#869,#873,#877,#881,#885,#889,#893,#897,#901,
#905,#909,#913,#917,#921,#925,#929,#933,#937,#941,#945,#949,#953,#957,
#961,#965,#969,#973,#977,#981,#985,#989,#993,#997,#1001,#1005,#1009,
#1013,#1017,#1021,#1025,#1029,#1033,#1037,#1041,#1045,#1049,#1053,#1057,
#1061,#1065,#1069,#1073,#1077,#1081,#1085,#1089,#1093,#1097,#1101));
#2404= ID_ATTRIBUTE('ALL',#2403);
#2406= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2397,#1145,.F.,
(#2407,#2417,#2421,#2425,#2429,#2433,#2437,#2441,#2445),
VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2407= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2408,$);
#2408= VOLUME_ELEMENT_LOCATION(#2409);
#2409= FEA_PARAMETRIC_POINT('sHexN_0',(0.,0.,0.));
#2411= REPRESENTATION('dpr0',(#2409,#2419,#2423,#2427,#2431,#2435,#2439,
#2443,#2447,#2452,#2457,#2461,#2465,#2469,#2474,#2479,#2483,#2487,#2491,
#2495,#2499),#2412);
#2412= PARAMETRIC_REPRESENTATION_CONTEXT('FEA parametric context',
'Parametric');
#2417= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2418,$);
#2418= VOLUME_ELEMENT_LOCATION(#2419);
#2419= FEA_PARAMETRIC_POINT('sHexN_1',(-1.,-1.,-1.));
#2421= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2422,$);
#2422= VOLUME_ELEMENT_LOCATION(#2423);
#2423= FEA_PARAMETRIC_POINT('sHexN_2',(1.,-1.,-1.));
#2425= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2426,$);
#2426= VOLUME_ELEMENT_LOCATION(#2427);
#2427= FEA_PARAMETRIC_POINT('sHexN_3',(1.,1.,-1.));
#2429= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2430,$);
#2430= VOLUME_ELEMENT_LOCATION(#2431);
#2431= FEA_PARAMETRIC_POINT('sHexN_4',(-1.,1.,-1.));
#2433= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2434,$);
#2434= VOLUME_ELEMENT_LOCATION(#2435);
#2435= FEA_PARAMETRIC_POINT('sHexN_5',(-1.,-1.,1.));
#2437= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
```

```
.UNSPECIFIED.),#2438,$);
#2438= VOLUME_ELEMENT_LOCATION(#2439);
#2439= FEA_PARAMETRIC_POINT('sHexN_6',(1.,-1.,1.));
#2441= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2442,$);
#2442= VOLUME_ELEMENT_LOCATION(#2443);
#2443= FEA_PARAMETRIC_POINT('sHexN_7',(1.,1.,1.));
#2445= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2446,$);
#2446= VOLUME_ELEMENT_LOCATION(#2447);
#2447= FEA_PARAMETRIC_POINT('sHexN_8',(-1.,1.,1.));
#2449= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2397,#1243,.F.,
(#2450,#2455,#2459,#2463,#2467),VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2450= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2451,$);
#2451= VOLUME_ELEMENT_LOCATION(#2452);
#2452= FEA_PARAMETRIC_POINT('sTetN_0',(0.,0.,0.));
#2455= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2456,$);
#2456= VOLUME_ELEMENT_LOCATION(#2457);
#2457= FEA_PARAMETRIC_POINT('sTetN_1',(-1.,-1.,-1.));
#2459= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2460,$);
#2460= VOLUME_ELEMENT_LOCATION(#2461);
#2461= FEA_PARAMETRIC_POINT('sTetN_2',(1.,-1.,-1.));
#2463= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2464,$);
#2464= VOLUME_ELEMENT_LOCATION(#2465);
#2465= FEA_PARAMETRIC_POINT('sTetN_3',(-1.,1.,-1.));
#2467= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2468,$);
#2468= VOLUME_ELEMENT_LOCATION(#2469);
#2469= FEA_PARAMETRIC_POINT('sTetN_4',(-1.,-1.,1.));
#2471= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2397,#1965,.F.,
(#2472,#2477,#2481,#2485,#2489,#2493,#2497),VOLUME_TENSOR2_3D_VARIABLE(
.STRESS.));
#2472= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2473,$);
#2473= VOLUME_ELEMENT_LOCATION(#2474);
#2474= FEA_PARAMETRIC_POINT('sWedN_0',(0.,0.,0.));
#2477= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2478,$);
#2478= VOLUME_ELEMENT_LOCATION(#2479);
#2479= FEA_PARAMETRIC_POINT('sWedN_1',(-1.,-1.,-1.));
#2481= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2482,$);
#2482= VOLUME_ELEMENT_LOCATION(#2483);
#2483= FEA_PARAMETRIC_POINT('sWedN_2',(1.,-1.,-1.));
#2485= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2486,$);
#2486= VOLUME_ELEMENT_LOCATION(#2487);
#2487= FEA_PARAMETRIC_POINT('sWedN_3',(-1.,1.,-1.));
#2489= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2490,$);
#2490= VOLUME_ELEMENT_LOCATION(#2491);
#2491= FEA_PARAMETRIC_POINT('sWedN_4',(-1.,-1.,1.));
#2493= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2494,$);
#2494= VOLUME_ELEMENT_LOCATION(#2495);
#2495= FEA_PARAMETRIC_POINT('sWedN_5',(1.,-1.,1.));
#2497= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2498,$);
#2498= VOLUME_ELEMENT_LOCATION(#2499);
#2499= FEA_PARAMETRIC_POINT('sWedN_6',(-1.,1.,1.));
#2501= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#2252,'STATIC STEP 2',2,#2255
,'',#2647);
#2502= SPECIFIED_STATE('Step 2 Base Specified State',
'Relating Specified State');
#2503= SPECIFIED_STATE('SPCCASE_12_22_2','Aggregator SPC 12 Step 2');
#2504= SPECIFIED_STATE('SPCVALSTATE_100_22','');
#2505= STATE_RELATIONSHIP(
```

```
'SPCVALSTATE_100_22 is related to SPCCASE_12_22_2','',#2503,#2504);
#2506= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2504,#2507,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2507= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.16',(#2501),#113,#35,(
#2266,#2269,#2270),'');
#2511= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2504,#2512,#2272,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2512= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.17',(#2501),#105,#35,(
#2270),'');
#2516= SPECIFIED_STATE('SPCVALSTATE_102_22','');
#2517= STATE_RELATIONSHIP(
'SPCVALSTATE_102_22 is related to SPCCASE_12_22_2','',#2503,#2516);
#2518= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2519,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2519= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.18',(#2501),#80,#35,(
#2266,#2269,#2270),'');
#2523= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2524,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2524= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.19',(#2501),#89,#35,(
#2266,#2269,#2270),'');
#2528= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2529,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2529= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.20',(#2501),#93,#35,(
#2266,#2269,#2270),'');
#2533= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2534,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2534= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.21',(#2501),#97,#35,(
#2266,#2269,#2270),'');
#2538= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2539,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2539= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.22',(#2501),#101,#35,(
#2266,#2269,#2270),'');
#2543= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2544,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2544= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.23',(#2501),#105,#35,(
#2266,#2269,#2270),'');
#2548= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2549,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2549= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.24',(#2501),#109,#35,(
#2266,#2269,#2270),'');
#2553= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2554,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2554= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.25',(#2501),#117,#35,(
#2266,#2269,#2270),'');
#2558= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2559,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2559= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.26',(#2501),#121,#35,(
#2266,#2269,#2270),'');
#2563= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2564,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2564= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.27',(#2501),#125,#35,(
#2266,#2269,#2270),'');
#2568= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2569,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2569= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.28',(#2501),#129,#35,(
#2266,#2269,#2270),'');
#2573= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2574,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
```

```
#2574= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.29',(#2501),#133,#35,(
#2266,#2269,#2270),'');
#2578= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2579,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2579= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.30',(#2501),#137,#35,(
#2266,#2269,#2270),'');
#2583= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2516,#2584,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2584= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.31',(#2501),#141,#35,(
#2266,#2269,#2270),'');
#2588= STATE_RELATIONSHIP(
'SPCCASE_12_22_2 is related to Step 2 Base Specified State','',#2502,
#2503);
#2589= LINEARLY_SUPERIMPOSED_STATE('LOADSTATECOMBINATION_22',
'Overall Factor Combined State');
#2590= STATE_COMPONENT('OverallComp','',#2589,1.);
#2592= SPECIFIED_STATE('LOADSTATECORE_300','Core Loads Specified State'
);
#2593= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#385,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-2.5),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2595= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#625,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-2.5),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2597= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#745,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-2.5),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2599= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#985,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-2.5),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2601= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#505,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2603= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#765,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2605= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#753,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2607= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#757,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2609= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#761,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2611= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#749,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2613= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#865,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2615= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#989,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2617= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#993,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2619= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#997,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2621= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#1001,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2623= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#1005,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2625= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_300',
'Item Factor State');
#2626= STATE_RELATIONSHIP('LOADSTATEITEM_300 is related to OverallComp',
```

```
'',#2590,#2625);
#2627= STATE_COMPONENT('ItemComp_22_300','',#2625,1.);
#2629= STATE_RELATIONSHIP(
'LOADSTATECORE_300 is related to ItemComp_22_300','',#2627,#2592);
#2630= SPECIFIED_STATE('LOADSTATECORE_400','Core Loads Specified State'
);
#2631= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#869,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2633= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#873,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2635= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#877,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2637= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#881,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2639= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#885,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2641= LINEARLY_SUPERIMPOSED_STATE('LOADSTATEITEM_400',
'Item Factor State');
#2642= STATE_RELATIONSHIP('LOADSTATEITEM_400 is related to OverallComp',
'',#2590,#2641);
#2643= STATE_COMPONENT('ItemComp_22_400','',#2641,1.);
#2645= STATE_RELATIONSHIP(
'LOADSTATECORE_400 is related to ItemComp_22_400','',#2643,#2630);
#2646= STATE_RELATIONSHIP(
'LOADSTATECOMBINATION_22 is related to Step 2 Base Specified State','',
#2502,#2589);
#2647= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 2','',
#2502);
#2648= OUTPUT_REQUEST_STATE('Step 2 Output Request State','',(#2501));
#2650= NODAL_FREEDOM_VALUES(#2648,#2403,#35,#2400,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#2652= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2648,#1145,.F.,
(#2653,#2655,#2656,#2657,#2658,#2659,#2660,#2661,#2662),
VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2653= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2408,$);
#2655= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2418,$);
#2656= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2422,$);
#2657= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2426,$);
#2658= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2430,$);
#2659= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2434,$);
#2660= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2438,$);
#2661= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2442,$);
#2662= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2446,$);
#2663= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2648,#1243,.F.,
(#2664,#2666,#2667,#2668,#2669),VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2664= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2451,$);
#2666= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2456,$);
#2667= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2460,$);
#2668= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2464,$);
#2669= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2468,$);
```

```
#2670= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2648,#1965,.F.,
(#2671,#2673,#2674,#2675,#2676,#2677,#2678),VOLUME_TENSOR2_3D_VARIABLE(
.STRESS.));
#2671= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2473,$);
#2673= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2478,$);
#2674= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2482,$);
#2675= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2486,$);
#2676= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2490,$);
#2677= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2494,$);
#2678= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2498,$);
#2679= CONTROL_LINEAR_STATIC_ANALYSIS_STEP(#2252,'STATIC STEP 3',3,#2255
,'',#2780);
#2680= SPECIFIED_STATE('Step 3 Base Specified State',
'Relating Specified State');
#2681= SPECIFIED_STATE('SPCCASE_12_23_3','Aggregator SPC 12 Step 3');
#2682= SPECIFIED_STATE('SPCVALSTATE_100_23','');
#2683= STATE_RELATIONSHIP(
'SPCVALSTATE_100_23 is related to SPCCASE_12_23_3','',#2681,#2682);
#2684= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2682,#2685,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2685= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.32',(#2679),#113,#35,(
#2266,#2269,#2270),'');
#2689= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2682,#2690,#2272,(
CONTEXT_DEPENDENT_MEASURE(0.)));
#2690= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_100.33',(#2679),#105,#35,(
#2270),'');
#2694= SPECIFIED_STATE('SPCVALSTATE_102_23','');
#2695= STATE_RELATIONSHIP(
'SPCVALSTATE_102_23 is related to SPCCASE_12_23_3','',#2681,#2694);
#2696= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2697,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2697= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.34',(#2679),#80,#35,(
#2266,#2269,#2270),'');
#2701= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2702,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2702= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.35',(#2679),#89,#35,(
#2266,#2269,#2270),'');
#2706= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2707,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2707= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.36',(#2679),#93,#35,(
#2266,#2269,#2270),'');
#2711= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2712,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2712= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.37',(#2679),#97,#35,(
#2266,#2269,#2270),'');
#2716= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2717,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2717= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.38',(#2679),#101,#35,(
#2266,#2269,#2270),'');
#2721= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2722,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2722= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.39',(#2679),#105,#35,(
#2266,#2269,#2270),'');
#2726= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2727,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2727= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.40',(#2679),#109,#35,(
```

```
#2266,#2269,#2270),'');
#2731= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2732,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2732= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.41',(#2679),#117,#35,(
#2266,#2269,#2270),'');
#2736= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2737,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2737= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.42',(#2679),#121,#35,(
#2266,#2269,#2270),'');
#2741= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2742,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2742= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.43',(#2679),#125,#35,(
#2266,#2269,#2270),'');
#2746= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2747,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2747= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.44',(#2679),#129,#35,(
#2266,#2269,#2270),'');
#2751= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2752,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2752= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.45',(#2679),#133,#35,(
#2266,#2269,#2270),'');
#2756= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2757,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2757= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.46',(#2679),#137,#35,(
#2266,#2269,#2270),'');
#2761= SINGLE_POINT_CONSTRAINT_ELEMENT_VALUES(#2694,#2762,#2262,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)));
#2762= SINGLE_POINT_CONSTRAINT_ELEMENT('SPC1_102.47',(#2679),#141,#35,(
#2266,#2269,#2270),'');
#2766= STATE_RELATIONSHIP(
'SPCCASE_12_23_3 is related to Step 3 Base Specified State','',#2680,
#2681);
#2767= LINEARLY_SUPERIMPOSED_STATE('LOADSTATECOMBINATION_23',
'Overall Factor Combined State');
#2768= STATE_COMPONENT('OverallComp','',#2767,1.);
#2770= STATE_RELATIONSHIP('LOADSTATEITEM_200 is related to OverallComp',
'',#2768,#2390);
#2771= STATE_COMPONENT('ItemComp_23_200','',#2390,1.);
#2772= STATE_RELATIONSHIP(
'LOADSTATECORE_200 is related to ItemComp_23_200','',#2771,#2357);
#2773= STATE_RELATIONSHIP('LOADSTATEITEM_300 is related to OverallComp',
'',#2768,#2625);
#2774= STATE_COMPONENT('ItemComp_23_300','',#2625,1.);
#2775= STATE_RELATIONSHIP(
'LOADSTATECORE_300 is related to ItemComp_23_300','',#2774,#2592);
#2776= STATE_RELATIONSHIP('LOADSTATEITEM_400 is related to OverallComp',
'',#2768,#2641);
#2777= STATE_COMPONENT('ItemComp_23_400','',#2641,1.);
#2778= STATE_RELATIONSHIP(
'LOADSTATECORE_400 is related to ItemComp_23_400','',#2777,#2630);
#2779= STATE_RELATIONSHIP(
'LOADSTATECOMBINATION_23 is related to Step 3 Base Specified State','',
#2680,#2767);
#2780= CONTROL_LINEAR_STATIC_LOAD_INCREMENT_PROCESS('STATIC STEP 3','',
#2680);
#2781= OUTPUT_REQUEST_STATE('Step 3 Output Request State','',(#2679));
#2783= NODAL_FREEDOM_VALUES(#2781,#2403,#35,#2400,(UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.),UNSPECIFIED_VALUE(
.UNSPECIFIED.),UNSPECIFIED_VALUE(.UNSPECIFIED.)));
#2785= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2781,#1145,.F.,
(#2786,#2788,#2789,#2790,#2791,#2792,#2793,#2794,#2795),
VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2786= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
```

```
.UNSPECIFIED.),#2408,$);
#2788= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2418,$);
#2789= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2422,$);
#2790= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2426,$);
#2791= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2430,$);
#2792= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2434,$);
#2793= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2438,$);
#2794= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2442,$);
#2795= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2446,$);
#2796= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2781,#1243,.F.,
(#2797,#2799,#2800,#2801,#2802),VOLUME_TENSOR2_3D_VARIABLE(.STRESS.));
#2797= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2451,$);
#2799= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2456,$);
#2800= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2460,$);
#2801= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2464,$);
#2802= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2468,$);
#2803= VOLUME_3D_ELEMENT_LOCATION_POINT_VARIABLE_VALUES(#2781,#1965,.F.,
(#2804,#2806,#2807,#2808,#2809,#2810,#2811),VOLUME_TENSOR2_3D_VARIABLE(
.STRESS.));
#2804= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2473,$);
#2806= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2478,$);
#2807= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2482,$);
#2808= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2486,$);
#2809= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2490,$);
#2810= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2494,$);
#2811= VOLUME_3D_ELEMENT_VALUE_AND_LOCATION(UNSPECIFIED_VALUE(
.UNSPECIFIED.),#2498,$);
#2812= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#745,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-31.25),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2814= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#841,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-31.25),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2816= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#985,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-31.25),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2818= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#1081,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-31.25),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2820= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#769,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2822= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#793,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2824= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#817,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2826= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#865,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2828= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#961,#35,#2359,(
```

```
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2830= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#1009,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2832= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#1033,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2834= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#1057,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-62.5),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2836= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#889,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2838= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#913,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2840= NODAL_FREEDOM_ACTION_DEFINITION(#2357,#937,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(-125.),CONTEXT_DEPENDENT_MEASURE(0.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2842= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#385,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-2.5),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2844= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#625,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-2.5),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2846= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#745,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-2.5),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2848= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#985,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-2.5),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2850= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#505,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2852= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#765,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2854= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#753,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2856= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#757,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2858= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#761,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2860= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#749,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2862= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#865,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2864= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#989,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2866= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#993,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2868= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#997,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2870= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#1001,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2872= NODAL_FREEDOM_ACTION_DEFINITION(#2592,#1005,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-5.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2874= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#869,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
```

```
#2876= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#873,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2878= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#877,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2880= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#881,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
#2882= NODAL_FREEDOM_ACTION_DEFINITION(#2630,#885,#35,#2359,(
CONTEXT_DEPENDENT_MEASURE(0.),CONTEXT_DEPENDENT_MEASURE(-10.),
CONTEXT_DEPENDENT_MEASURE(0.)),.APPLIED_LOADS.);
ENDSEC;

END-ISO-10303-21;
```

END OF DOCUMENT