Generic Neutral Files

CAESAR II Neutral File Interface

The general neutral file can be used to send data either in to or out of the standard **CAESAR II** binary input file, otherwise known as the _A file. The name of the file used or generated by this interface is the **CAESAR II** jobname with the extension .CII. The intent of this interface is to allow users access to any particular data item from an _A input file, to enable a complete _A file to be built from a CAD program, and to allow **CAESAR II** input data to be used for other analysis purposes.

Users implementing this interface should be warned that the content and format described in this section is subject to change, as a function of the enhancements made to the **CAESAR II** program. Every effort will be made to keep such "drastic" changes to a minimum.

The **CAESAR II** neutral file, henceforth referred to as the .CII file, is divided into sections which organize the piping data in logical groupings. Each major section is discussed below. Details of each item are discussed to the right of the page. Section divisions are denoted in the neutral file by the '#$' character sequence found in columns 1 & 2. The token following the '#$' character sequence is a section identifier, used by the program for data sequencing purposes, and to aid the user in reading the neutral file.

Several third-party CAD programs, such as AVEVA's PDMS and Jacobus' Plant Space also support this neutral file.

For each item listed on the following pages, the necessary FORTRAN format for the input/output is provided. The following variables are used in dimensioning arrays:

- N1—Base memory allocation quantity, used to set array sizes. For example, if N1=2,000, your neutral file can handle up to 2,000 elements.
- N2—1/2 N1
- N3—1/3 N1
- N4—1/4 N1
- N5—1/5 N1
- N6—N1/13.33

Version and Job Title Information

#$ VERSION. This is the section division header. The **#$** and space are required, as well as the word **VERSION**, all in capital letters.

Use FORTRAN format (2X, 4G13.6) to write the values of the following variables on the first line of the neutral file:

- GVERSION is the version of the neutral file interface being used. This corresponds to the major version number of **CAESAR II**, i.e. 4 for 4.x.
- RVERSION is the specific **CAESAR II** version generating this file, i.e. 4.50.
- SPARE are unused (at this time) locations on the record.

The next 60 lines of 75 characters each are reserved for the **CAESAR II** title-page text. Use FORTRAN format (2X, A75).

The last line of the job title array, if found to be blank, is set by this transfer program. The text that is set here indicates that the file was created by this interface.

Control Information

#$ CONTROL. This is the section division header. The **#$** and space are required, as well as the word **CONTROL**, in capital letters.

Use FORTRAN format (2X, 4I13) to write the values of the following variables on the next line of the neutral file:

- NUMELT is the number of piping elements (spreadsheets) in the input file.
- NUMNOZ is the number of nozzles in the input file.
- NOHGRS is the number of spring hangers in the input file.
- NONAM is the number of Node Name data blocks in the input file.
- NORED is the number of reducers in the input file.

Next, write 11-members of the array (IAUXAU) that contains the number of auxiliary data types used in the input file, followed by the vertical axis indicator. Use FORTRAN format (2X, 6I13).

These 11 values from the IAUXAU array are the following:

1 The number of bend auxiliary data blocks in the input file.

2 The number of rigid-element auxiliary data blocks in the input file.

3 The number of expansion-joint auxiliary data blocks in the input file.

4 The number of restraint auxiliary data blocks in the input file.

5 The number of displacement auxiliary data blocks in the input file.

6 The number of force/moment auxiliary data blocks in the input file.

7 The number of uniform-load auxiliary data blocks in the input file.

8 The number of wind-load auxiliary data blocks in the input file.

9 The number of element-offset auxiliary data blocks in the input file.

10 The number of allowable-stress auxiliary data blocks in the input file.

11 The number of intersection auxiliary data blocks in the input file.

- IZUP flag. Equal to 0 for the global -Y axis vertical, equal to 1 for the global -Z axis vertical.

Basic Element Data

#$ ELEMENTS. This is the section division header. The **#$** and space are required, as well as the word **ELEMENTS**, all in capital letters.

This section of the file contains *integer* and *real* data for each element in the input file. The data are organized as such:

1 *real* values for element "i"

2 *integer* values for element "i"

3 *real* values for element "i+1"

4 *integer* values for element "i+1"

These *real* and *integer* values are stored in arrays, described as follows:

A 36-member array (REL) contains the *real* basic-element data. The REL array is dimensioned (N1,36).

Use FORTRAN format (2X, 6G13.6) to write the values of the following 36 items on the appropriate six lines of the neutral file.

1   FROM node number

2   TO node number

3   Delta X

4   Delta Y

5   Delta Z

6   Diameter (value stored here is actual OD)

7   Wall Thickness (actual)

8   Insulation Thickness

9   Corrosion Allowance

10 Thermal Expansion Coefficient #1 (or Temperature #1)

11 Thermal Expansion Coefficient #2 (or Temperature #2)

12 Thermal Expansion Coefficient #3 (or Temperature #3)

13 Thermal Expansion Coefficient #4 (or Temperature #4)

14 Thermal Expansion Coefficient #5 (or Temperature #5)

15 Thermal Expansion Coefficient #6 (or Temperature #6)

16 Thermal Expansion Coefficient #7 (or Temperature #7)

17 Thermal Expansion Coefficient #8 (or Temperature #8)

18 Thermal Expansion Coefficient #9 (or Temperature #9)

19 Pressure #1

20 Pressure #2

21 Pressure #3

22 Pressure #4

23 Pressure #5

24 Pressure #6

25 Pressure #7

26 Pressure #8

27 Pressure #9

28 Elastic Modulus

29 Poisson's Ratio

30 Pipe Density

31 Insulation Density

32 Fluid Density

33 Plus Mill Tolerance

34 Minus Mill Tolerance

35 Seam Weld (1=Yes, 0=No)

36 Hydro Pressure

Non-specified *real* values are assigned a value of 0.0 by this interface.

If the delta coordinates are not specified, they default to zero. If the **To/From** fields are not specified, it is considered an error.

A 18-member array (IEL) contains the pointers to the auxiliary data arrays. The IEL array is dimensioned (N1,18). **NOTE, at this time, only 13 of the members of this array are utilized!**

Use FORTRAN format (2X, 6I13) to write the values of the following 13 items on the next two lines of the neutral file.

1 Pointer to **Bend Auxiliary** field. This indicates where in the bend auxiliary array the bend data for the current element can be found.

2 Pointer to **Rigid Element Auxiliary** field.

3 Pointer to **Expansion Joint Auxiliary** field.

4 Pointer to **Restraint Auxiliary** field.

5 Pointer to **Displacement Auxiliary** field.

6 Pointer to **Force/Moment Auxiliary** field.

7 Pointer to **Uniform Load Auxiliary** field.

8 Pointer to **Wind Load Auxiliary** field.

9 Pointer to **Element Offset Auxiliary** field.

10 Pointer to **Allowable Stress Auxiliary** field.

11 Pointer to **Intersection Auxiliary** field.

12 Pointer to **Node Name Auxiliary** field.

13 Pointer to **Reducer Auxiliary** field.

A pointer value of zero should be used where there is no auxiliary data of a particular type associated with the current element.

Auxiliary Element Data

#$ AUX_DATA. This is the section division header. The **#$** and space are required, as well as the word **AUX_DATA**, all in capital letters.

This section of the file contains the auxiliary data corresponding to the elements. This data is arranged in the same order as the IAUXAU array described previously. For example, if IAUXAU(1) contains a 3, then there are 3 bends in the model, and their data is found next in the neutral file. Also assume that IAUXAU(2) contains a 5, then there are 5 rigid elements in the model and their data follows the bend data.

Each set of auxiliary data is separated by a sub-section header. If a particular value in IAUXAU is zero, then only the subsection header is written to the neutral file.

The data storage for these arrays is allocated at run time, based on the available free system memory. These arrays are allocated proportionally, as a percentage of the n-umber of elements allowed. Four proportions are used: 1/2, 1/3, 1/4, and 1/5. These proportions correspond to the variables: N2, N3, N4, and N5. Maintaining these proportions ensures that the neutral file reader can accept the file.

#$ NODENAME. This is the subsection header that defines the start of Node Name data. (In order to maintain downward compatibility, this section is optional.) The data for each element set of node names in the input file is listed here.

A two-member array (NAM) defines each set of node names. The NAM array is dimensioned (N6, 2). Use FORTRAN format (2X, A10, 16X, A10) to read first the character name of the **FROM** node and then that of the **TO** node.

#$ BEND. This is the subsection header that defines the start of the bend data. The data for each bend in the input file is listed here.

An 11-member array (BND) defines each bend. The BND array is dimensioned (N3,11).

Use FORTRAN format (2X, 6G13.6) to write the values of the following 11 items on the next two lines of the neutral file.

1   bend radius

2   type: 1 - single flange 2 - double flange 0 or blank - welded

3   angle to node position #1

4   node number at position #1

5   angle to node position #2

6   node number at position #2

7   angle to node position #3

8   node number at position #3

9   number of miter cuts

10 fitting thickness of bend if different from the pipe

11 Seam Weld (1=Yes, 0=No)

12 Bend flexibility (K) factor

#$ RIGID. This is the subsection header that defines the start of the rigid data. The data for each rigid in the input file is listed here.

A single-element array (RIG) for each rigid. The RIG array is dimensioned (N3,1). The single element of the array represents the rigid weight.

Use FORTRAN format (2X, 6G13.6) to write the value.

#$ EXPJT. This is the subsection header that defines the start of the expansion joint data. The data for each expansion joint in the input file is listed here. The EXP array is dimensioned (N5,5).

Use FORTRAN format (2X, 6G13.6) to write the values of the following five items on the next line of the neutral file.

1   axial stiffness

2   transverse stiffness

3   bending stiffness

4   torsional stiffness

5   effective inside bellows diameter

#$ RESTRANT. This is the subsection header that defines the start of the restraint data.

The data for each restraint auxiliary data block in the input file is listed here. The RES array is dimensioned (N2,36).

Use FORTRAN format (2X, 6G13.6) to write the values of the following nine items on the next two lines of the neutral file. These nine items are repeated four times for the four possible restraints defined in the auxiliary data block.

This will require two lines in the neutral file for each restraint specification, which means eight lines total for each restraint auxiliary.

1   restraint node number

2   restraint type (see additional notes to follow)

3   restraint stiffness

4   restraint gap

5   restraint friction coefficient

6   restraint connecting node

7   X direction cosine

8   Y direction cosine

9   Z direction cosine

Note: Items 3-9 may change based on the value of the restraint type. For more information refer to the help text for more information.

The restraint type is an integer value whose valid range is from 1 to 62.

The 62 possible restraint types are

1 - 5	ANC	X	Y	Z	RX
6 - 10	RY	RZ	GUI	LIM	XSNB
11 - 15	YSNB	ZSNB	+X	+Y	+Z
16 - 20	-X	-Y	-Z	+RX	+RY
21 - 25	+RZ	-RX	-RY	-RZ	+LIM
26 - 30	-LIM	XROD	YROD	ZROD	+XROD
31 - 35	+YROD	+ZROD	-XROD	-YROD	-ZROD
36 - 40	X2	Y2	Z2	RX2	RY2
41 - 45	RZ2	+X2	+Y2	+Z2	-X2
46 - 50	-Y2	-Z2	+RX2	+RY2	+RZ2
51 - 55	-RX2	-RY2	-RZ2	XSPR	YSPR
56 - 60	ZSPR	+XSNB	+YSNB	+ZSNB	-XSNB
61 - 62	-YSNB	-ZSNB			

#$ DISPLMNT. This is the subsection header that defines the start of the displacement data.

The data for each displacement auxiliary data block in the input file is listed here.

Use FORTRAN format (2X, 6G13.6) to write the values of the following 55 items on the next lines of the neutral file. The DIS array is dimensioned (N3,110).

This will require ten lines in the neutral file for each displacement specification, which means 20 lines total for each displacement auxiliary.

```
                      1) first displacement node number
{vector 1}   2) DX        3) DY       4) DZ       5) RX       6) RY       7) RZ
{vector 2}   8) DX        9) DY       10) DZ      11) RX      12) RY      13) RZ
{vector 3}   14) DX       15) DY      16) DZ      17) RX      18) RY      19) RZ
{vector 4}   20) DX       21) DY      22) DZ      23) RX      24) RY      25) RZ
{vector 5}   26) DX       27) DY      28) DZ      29) RX      30) RY      31) RZ
{vector 6}   32) DX       33) DY      34) DZ      35) RX      36) RY      37) RZ
{vector 7}   38) DX       39) DY      40) DZ      41) RX      42) RY      43) RZ
{vector 8}   44) DX       45) DY      46) DZ      47) RX      48) RY      49) RZ
{vector 9}   50) DX       51) DY      52) DZ      53) RX      54) RY      55) RZ


                      56) second displacement node number
{vector 1}   57) DX       58) DY      59) DZ      60) RX      61) RY      62) RZ
{vector 2}   63) DX       64) DY      65) DZ      66) RX      67) RY      68) RZ
{vector 3}   69) DX       70) DY      71) DZ      72) RX      73) RY      74) RZ
{vector 4}   75) DX       76) DY      77) DZ      78) RX      79) RY      80) RZ
{vector 5}   81) DX       82) DY      83) DZ      84) RX      85) RY      86) RZ
{vector 6}   87) DX       88) DY      89) DZ      90) RX      91) RY      92) RZ
{vector 7}   93) DX       94) DY      95) DZ      96) RX      97) RY      98) RZ
{vector 8}   99) DX       100) DY     101) DZ     102) RX     103) RY     104) RZ
{vector 9}   105) DX      106) DY     107) DZ     108) RX     109) RY     110) RZ
```

These 55 items are repeated 2 times for the two possible displacements defined on the auxiliary.

Note: Unspecified displacement values (i.e., free-displacement degrees of freedom) are designated through the use of a value of 9999.99.

#$ FORCMNT. This is the subsection header that defines the start of the force/moment data. The data for each force/moment auxiliary data block in the input file is listed here.

Use FORTRAN format (2X, 6G13.6) to write the values of the following 55 items on the next ten lines of the neutral file. The FOR array is dimensioned (N3,38).

This will require ten lines in the neutral file for each force/moment specification, which means 20 lines total for each force/moment auxiliary data block.

	1) first force/moment node number					
{vector 1}	2) FX	3) FY	4) FZ	5) MX	6) MY	7) MZ
{vector 2}	8) FX	9) FY	10) FZ	11) MX	12) MY	13) MZ
{vector 3}	14) FX	15) FY	16) FZ	17) MX	18) MY	19) MZ
{vector 4}	20) FX	21) FY	22) FZ	23) MX	24) MY	25) MZ
{vector 5}	26) FX	27) FY	28) FZ	29) MX	30) MY	31) MZ
{vector 6}	32) FX	33) FY	34) FZ	35) MX	36) MY	37) MZ
{vector 7}	38) FX	39) FY	40) FZ	41) MX	42) MY	43) MZ
{vector 8}	44) FX	45) FY	46) FZ	47) MX	48) MY	49) MZ
{vector 9}	50) FX	51) FY	52) FZ	53) MX	54) MY	55) MZ

	56) second displacement node number					
{vector 1}	57) FX	58) FY	59) FZ	60) MX	61) MY	62) MZ
{vector 2}	63) FX	64) FY	65) FZ	66) MX	67) MY	68) MZ
{vector 3}	69) FX	70) FY	71) FZ	72) MX	73) MY	74) MZ
{vector 4}	75) FX	76) FY	77) FZ	78) MX	79) MY	80) MZ
{vector 5}	81) FX	82) FY	83) FZ	84) MX	85) MY	86) MZ
{vector 6}	87) FX	88) FY	89) FZ	90) MX	91) MY	92) MZ
{vector 7}	93) FX	94) FY	95) FZ	96) MX	97) MY	98) MZ
{vector 8}	99) FX	100) FY	101) FZ	102) MX	103) MY	104) MZ
{vector 9}	105) FX	106) FY	107) FZ	108) MX	109) MY	110) MZ

#$ UNIFORM. This is the subsection header that defines the start of the uniform load data. The data for each uniform load in the input file is listed here.

Use FORTRAN format (2X, 6G13.6) to write the values of the following 9 items on the next two lines of the neutral file. The UNI array is dimensioned (N5,9).

This will require two lines in the neutral file for each uniform load auxiliary data block.

{vector 1 & 2}   UX1   UY1 UZ1   UX2 UY2 UZ2

{vector 3}   UX3   UY3 UZ3

#$ WIND. This is the subsection header that defines the start of the wind/wave data. The data for each wind/wave specification in the input file is listed here. The WIND array is dimensioned (N5,5).

Use FORTRAN format (2X, 6G13.6) to write the set of values on the next line of the neutral file. This will require a single line in the neutral file for each wind auxiliary.

The five data items on each line are as follows:

1   entry type (0.0 for Wind, 1.0 for Wave, 2.0 for Off)

2   wind shape factor or wave drag coefficient

3   wave added mass coefficient

4   wave lift coefficient

5   wave marine growth

#$ OFFSETS. This is the subsection header that defines the start of the element offset data. The data for each offset pipe in the input file is listed here.

Use FORTRAN format (2X, 6G13.6) to write the values of the following six items on the next line of the neutral file. The OFF array is dimensioned (N5,6).

This will require a single line in the neutral file for each offset auxiliary.

1   element FROM node offset in X direction

2   element FROM node offset in Y direction

3   element FROM node offset in Z direction

4   element TO node offset in X direction

5   element TO node offset in Y direction

6   element TO node offset in Z direction

#$ ALLOWBLS. This is the subsection header that defines the start of the allowable stress data. The data for each allowable spec in the input file is listed here.

Use FORTRAN format (2X, 6G13.6) to write the values of the following 108 items on the next eighteen lines of the neutral file. The ALL array is dimensioned (N5,108).

This will require eighteen lines in the neutral file for each allowable auxiliary.

1   cold allowable stress

2   hot allowable for thermal case #1

3   hot allowable for thermal case #2

4   hot allowable for thermal case #3

5   code cyclic reduction factor for thermal case #1

6   code cyclic reduction factor for thermal case #2

7   code cyclic reduction factor for thermal case #3

8   Eff.

9 S_y

10 fac

11 Pmax

12 piping code id

13 hot allowable for thermal case #4

14 hot allowable for thermal case #5

15 hot allowable for thermal case #6

16 hot allowable for thermal case #7

17 hot allowable for thermal case #8

18 hot allowable for thermal case #9

19 code cyclic reduction factor for thermal case #4

20 code cyclic reduction factor for thermal case #5

21 code cyclic reduction factor for thermal case #6

22 code cyclic reduction factor for thermal case #7

23 code cyclic reduction factor for thermal case #8

24 code cyclic reduction factor for thermal case #9

25 cycles for BW (butt-weld) fatigue pair #1

26 cycles for BW fatigue pair #2

27 cycles for BW fatigue pair #3

28 cycles for BW fatigue pair #4

29 cycles for BW fatigue pair #5

30 cycles for BW fatigue pair #6

31 cycles for BW fatigue pair #7

32 cycles for BW fatigue pair #8

33 stress for BW fatigue pair #1

34 stress for BW fatigue pair #2

35 stress for BW fatigue pair #3

36 stress for BW fatigue pair #4

37 stress for BW fatigue pair #5

38 stress for BW fatigue pair #6

39 stress for BW fatigue pair #7

40 stress for BW fatigue pair #8

41 cycles for FW (fillet-weld) fatigue pair #1

42 cycles for FW fatigue pair #2

43 cycles for FW fatigue pair #3

44 cycles for FW fatigue pair #4

45 cycles for FW fatigue pair #5

46 cycles for FW fatigue pair #6

47 cycles for FW fatigue pair #7

48 cycles for FW fatigue pair #8

49 stress for FW fatigue pair #1

50 stress for FW fatigue pair #2

51 stress for FW fatigue pair #3

52 stress for FW fatigue pair #4

53 stress for FW fatigue pair #5

54 stress for FW fatigue pair #6

55 stress for FW fatigue pair #7

56 stress for FW fatigue pair #8

Items 57 through 108 are for the TD/12 piping code.

Some of these items (notably 8-24) may have various meanings based on the active piping code.

#$ SIF&TEES. This is the subsection header that defines the start of the SIF/TEE data. The data for each SIF/TEE spec in the input file is listed here.

Use FORTRAN format (2X, 6G13.6) to write the values of the following 30 items, for each of the two tees that can be specified on the dialog. The SIF array is dimensioned (N4,60).

This will require five lines in the neutral file for each SIF/TEE specified, which means ten lines total for each auxiliary.

1   intersection node number

2   intersection type code, if not specified this auxiliary is only used to specify SIFs

3   SIF, in plane

4   SIF, out of plane

5   Weld id

6   Fillet

7   Pad thk

8   FTG Ro

9   crotch

10 weld id

11 B1

12 B2

Items 13 - 30 are for the TD/12 piping code.

#$ REDUCERS. This is the subsection header that defines the start of the REDUCER data. The data for each REDUCER spec in the input file is listed here.

Use FORTRAN format (2X, 6G13.6) to write the values of the following 5 items on the next line of the neutral file. The RED array is dimensioned (N6,5).

This will require one line in the neutral file for each REDUCER specified.

1   2nd diameter of the reducer

2   2nd thickness of the reducer

3   alpha angle of the reducer

4   R1 value of the reducer for the TD/12 piping code

5   R2 value of the reducer for the TD/12 piping code

These values are repeated for the second intersection specification.

Miscellaneous Data Group #1

#$ MISCEL_1. This is the section division header. The **#$** and space are required, as well as the word **MISCEL_1**, all in capital letters.

The data in this group consists of the material id (RRMAT) for each element in the input file, the nozzle data (VFLEX), the hanger data, and the execution options.

Material ID. The first array in this section (RRMAT) contains the material id number for each element in the input file. Use FORTRAN format (2X, 6G13.6).

The RRMAT array is dimensioned (N1). The material ids range from 1 to 699 (See the User's Guide for details).

The number of lines required to write the RRMAT array in the neutral file is determined by the following FORTRAN routine:

NLINES = NUMELT / 6

IF(MOD(NUMELT,6) .NE. 0) THEN
NLINES = NLINES + 1

ENDIF

Nozzles. The next set of data describes the flexible (WRC-297, PD-5500, API 650) nozzles in the input file.

Note:	9999.99 represents infinity.

Use FORTRAN format (2X, 6G13.6). The nozzle (VFLEX) contains 16 values for each nozzle in the input.

This will require four lines in the neutral WRC-297, PD-5500, and/or API 650 spreadsheet. The VFLEX array is dimensioned (N6, 16).

For WRC-297 nozzles, the 16 items are

1   Nozzle Node Number

2   Vessel Node Number (optional)

3   Nozzle type indicator (-1.0101 = 297, 1.0 = 650)

4   Nozzle Outside Diameter (in.)

5   Nozzle Wall Thickness (in.)

6   Vessel Outside Diameter (in.)

7   Vessel Wall Thickness (in.)

8   Vessel Reinforcing Pad Thickness (in.)

9   Spare (not used)

10 Dist. to stiffeners or head (in.) (9999.99 = ì)

11 Dist. to opposite side stiffeners or head (in.) (9999.99 = ì)

12 Vessel centerline direction vector X

13 Vessel centerline direction vector Y

14 Vessel centerline direction vector Z

15 Vessel Temperature (optional) (°F)

16 Vessel Material # (optional)(1-17)

For PD-5500 nozzles, the 16 items are

1   Nozzle Node Number

2   Vessel Node Number (optional)

3   Nozzle type indicator (2.0-5500)

4 Vessel Type (0-Cylinder, 1-Sphere)

5 Nozzle Outside Diameter (in.)

6 Vessel Outside Diameter (in.)

7 Vessel Wall Thickness (in.)

8 Vessel Reinforcing Pad Thickness (in.)

9 Spare (not used)

10 Dist. to stiffeners or head (in.) (9999.99 = ì)

11 Dist. to opposite side stiffeners or head (in.) (9999.99 = ì)

12 Vessel centerline direction vector X

13 Vessel centerline direction vector Y

14 Vessel centerline direction vector Z

15 Vessel Temperature (optional) (°F)

16 Vessel Material # (optional) (1-17)

For API 650 nozzles, the 16 items are

1 Nozzle Node Number

2 Vessel Node Number (optional)

3 Nozzle type indicator (1.0 = 650)

4 Nozzle Outside Diameter (in.)

5 Nozzle Wall Thickness (in.)

6 Vessel Outside Diameter (in.)

7 Vessel Wall Thickness (in.)

8 Spare (not used)

9 Reinforcing on 1 - shell, or 2 - nozzle

10 Height of nozzle centerline (in.)

11 Height of tank fluid (in.)

12 Not Used

13 Specific gravity of fluid

14 Thermal expansion coefficient (in/in/deg)

15 Delta Temperature (°F)

16 Elastic Modulus (psi)

Hangers. The next set of data describes the spring hangers in the input file. Some of the hanger data listed below represent uninitialized data. In the instances where this uninitialized data represent infinite values (such as maximum travel limit and available space) it is reported here as 9999.99.

The next line contains values for the following parameters in FORTRAN format (2X, I13, 5G13.6):

- IDFTABLE is the default hanger table.
- DEFVAR is the default for allowed load variation.
- DEFRIG is the default for rigid support displacement criteria.
- DEFMXTRAVEL is the default for maximum allowed travel.
- DEFSHTSPR is the default for allowing short range springs (0=no 1=yes).
- DEFMUL is the default multi load case design option.

The next line contains values for the following parameters in FORTRAN format (2X, 5I13):

- IDFOPER is the default # of hanger design operating cases (always 1)
- IACTCLD is the default cold load calculation switch (0=no, 1=yes).
- IHGRLDS is the number of hanger operating loads (0 -3).
- IACTUAL is the load case defining actual cold loads.
- IMULTIOPTS is the multiple load case design option (1-7).

An array of hanger node numbers (IHGRNODE) is read/written for each hanger in the input file and is dimensioned (N5). There will be seven lines in the neutral file for this data, if all N5 hangers are specified. Use FORTRAN format (2X, 6I13).

A 10-element array (HGRDAT) is read/written for each hanger in the input file. The HGRDAT array is dimensioned (10,N5). Each hanger in the model will require two lines in the neutral file. Use FORTRAN format (2X, 6G13.6).

1 hanger stiffness

2 allowable load variation

3 rigid support displacement criteria

4 allowed space for hanger

5 cold load #1 (theoretical)

6 hot load #1 (initialize to 0.0)

7 user defined operating load f/ variable springs (init to 0.0)

8 maximum allowed travel limit

9 multiple load case design option

10 hanger constant effort support load

A four-element array (IHGRFREE) is read/written for each hanger in the input file. The IHGRFREE array is dimensioned (4,N5). Each hanger in the file will require one line in the neutral file.

Use FORTRAN format (2X, 6I13).

1 anchor node to be freed (#1)

2 anchor node to be freed (#2)

3 d.o.f. type for #1 (1-free Y, 2-free XY, 3-free ZY, 4-free X, Y, Z, 5-free all)

4 d.o.f. type for #2

An array (IHGRNUM) lists the number of hangers at this location, for each hanger in the input file. There will be one entry here for every hanger in the file. The IHGRNUM array is dimensioned (N5). There will be seven lines in the neutral file for this data, if all N5 hangers are specified. Use FORTRAN format (2X, 6I13).

An array (IHGRTABLE) listing the hanger table numbers for each hanger in the input file. There will be one entry here for every hanger in the file. The IHGRTABLE is dimensioned (N5). There will be seven lines in the neutral file for this data, if all N5 hangers are specified. Use FORTRAN format (2X, 6I13).

An array of flags (IHGRSHORT) indicates if short range springs can be used at each hanger location. The IHGRSHORT array is dimensioned (N5). There will be seven lines in the neutral file for this data. Use FORTRAN format (2X, 6I13).

- 0 = can't use short range springs
- 1 = can use short range springs

An array of connecting node numbers (IHGRCN) is available for each hanger. The IHGRCN array is dimensioned (N5). There will be seven lines in the neutral file for this data, if all N5 hangers are specified. Use FORTRAN format (2X, 6I13).

Execution Options. The next section of data defines the execution options used by the program. Use FORTRAN format (2X, 4I13, G13.6, I13). This will require three lines in the neutral file. These values are

- Print forces on rigids and expansion joints 0=no, 1=yes
- Print alphas & pipe props. during error checking 0=no, 1=yes
- Activate Bourdon Pressure Effects 0, 1, or 2
- Activate Branch Error and Coordinate Prompts 0=no, 1=yes
- Thermal Bowing Delta Temperature degrees
- Use Liberal Stress Allowable 0=no, 1=yes

For the following data, use FORTRAN format: (2X, I13, 2G13.6, 3I13):

- Uniform Load Input in g''s 0=no, 1=yes
- Stress Stiffening due to Pressure 0, 1, 2
- Ambient Temperature (If not 70.00 deg F) degrees
- FRP Expansion * 1,000,000 len/len/deg
- Optimizer 0-Both, 1-CuthillMcKee, 2-Collins
- Next Node Selection 0-Decreasing, 1-Increasing

For the following data, use FORTRAN format (2X, 4I13, G13.6, I13):

- Final Ordering 0-Reversed, 1-Not Reversed
- Collins Ordering 0-Band, 1-No. of Coefficients
- Degree Determination 0-Connections, 1-Band
- User Control 0-None, 1-Allow User Re-Looping
- FRP Shear ratio
- Laminate type

Units Conversion Data

#$ UNITS. This is the section division header. The **#$** and space are required, as well as the word **UNITS**. The data in this section defines both the conversion constants as well as the conversion labels. The conversion constants are all **REAL*4** values in FORTRAN format (2X, 6G13.6). This will require four lines in the neutral file. The following are character definitions for the labels:

- CNVLEN is the length conversion.
- CNVFOR is the force conversion.
- CNVMAS is the mass conversion.
- CNVMIN is the moment (input) conversion.
- CNVMOU is the moment (output) conversion.
- CNVSTR is the stress conversion.
- CNVTSC is the temperature conversion.
- CNVTOF is the temperature offset.
- CNVPRE is the pressure conversion.
- CNVYM is Young's modulus conversion.
- CNVPDN is the pipe density conversion.
- CNVIDN is the insulation density conversion.
- CNVFDN is the fluid density conversion.
- CNVTSF is the translational stiffness conversion.
- CNVUNI is the uniform load conversion.
- CNVWND is the wind load conversion
- CNVELE is the elevation conversion
- CNVCLN is the compound length conversion
- CNVDIA is the diameter conversion
- CNVTHK is the wall thickness conversion

Next, enter the following units' labels, one per line, in the format given in the label descriptions. This will require 24 lines in the neutral file.

CCVNAME - name of the units used, i.e. english, si, ..(CHARACTER*15)

CCVNOM - "on" or "off" and tells PREPIP whether or not nominal diameters are allowed (CHARACTER* 3).

CCVLEN - length label (CHARACTER* 3)

CCVFOR - force label (CHARACTER* 3)

CCVMAS - mass label (CHARACTER* 3)

CCVMIN - moment (input) label (CHARACTER* 6)

CCVMOU - moment (output) label (CHARACTER* 6)

CCVSTR - stress label (CHARACTER*10)

CCVTSC - temperature label (CHARACTER* 1)

CCVTOF - temperature offset/label (CHARACTER* 1)

CCVPRE - pressure label (CHARACTER*10)

CCVYM - young's modulus label (CHARACTER*10)

CCVPDN - pipe density label (CHARACTER*10)

CCVIDN - insulation density label (CHARACTER*10)

CCVFDN - fluid density label (CHARACTER*10)

CCVTSF - translational stiffness label (CHARACTER* 7)

CCVRSF - rotational stiffness label (CHARACTER*10)

CCVUNI - uniform load label (CHARACTER* 7)

CCVGLD - gravitional load label (CHARACTER* 3)

CCVWND - wind load label (CHARACTER*10)

CCVELE - elevation label (CHARACTER* 3)

CCVCLN - compound length label (CHARACTER* 3)

CCVDIA - diameter label (CHARACTER* 3)

CCVTHK - wall thickness label (CHARACTER* 3)

Nodal Coordinate Data

#$ COORDS . This is the section division header. The **#$** and space are required, as well as the word **COORDS**, all in capital letters. This section only exists in Versions 3.22 and later.

The data in this section of the neutral file is optional; it may not exist. The existence of this data depends on the user's preference and the particular job. This section of the neutral file is used to specify the X, Y, Z global coordinates of the starting node point of each discontinuous piping segment. This data, if it exists, is defined below.

The NXYZ value defines how many sets of coordinates follow. Use FORTRAN format (2X, I13).

INODE, XCORD, YCORD, ZCORD This line of four values is repeated NXYZ times.

Use FORTRAN format (2X, I13, 3F13.4) to define a node number and its X, Y, Z global coordinates.

Data Matrix Interface

CAESAR II offers an alternative neutral file, the Data Matrix Interface.

The generic **CAESAR II** data matrix input routine creates a **CAESAR II** file from a simple neutral file. It expects to read a file that contains a single line of data for each pipe in the model. Each line of data contains twelve parameters as follows:

ELMT N1   N2   DX   DY   DZ   DIAM THK   ANCH BEND BRAD RIGID

Where:

ELMT is the element number, sequential from 1.

N1 is the "from" node number.

N2 is the "to" node number.

DX is the delta dimension in the global "X" direction.

DY is the delta dimension in the global "Y" direction (the "Y" axis is vertical in **CAESAR II**).

DZ is the delta dimension in the global "Z" direction.

DIAM is the actual pipe diameter.

THK is the actual pipe wall thickness.

ANCH is a restraint flag, 1 if the "from" node is restrained, 0 otherwise. Currently ignored.

BEND is a bend indicator, 1 if the element has a bend at the "to" node, 0 otherwise.

BRAD is the bend radius if not a long radius bend.

Important: RIGID is a rigid element flag, 1 if the element is rigid, 0 otherwise.

All values in the matrix should be "real," floating point numbers. The format for each line of data should be (12E13.6). This generic interface does prompt for an arbitrary conversion constant for the delta dimensions and the diameter /thickness values to overcome any differences between the assumed units of the neutral file and the **CAESAR II** defaults.

Users developing an interface from scratch are urged to use the *Complete Neutral File* interface discussed in the next section. The *Data Matrix Interface* discussed above transfers the piping geometry only, which requires the analyst to input additional data to complete the stress model.